NASA CR-159,083

NASA Contractor Report 159083

NASA-CR-159083

1981 0014191

NLSCIDNT USER'S GUIDE

MAXIMUM LIKELIHOOD PARAMETER IDENTIFICATION COMPUTER PROGRAM WITH NONLINEAR ROTORCRAFT MODEL

SYSTEMS CONTROL, INC. (Vt)
1801 Page Mill Road
Palo Alto, California 94304

NASA Contract NAS1-14549
November 1979

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

TABLE OF CONTENTS

N81-22724#

I. INTRODUCTION AND OVERVIEW

System identification technology has been used successfully for many vehicles. Because of their large number of degrees of freedom and complex aerodynamic interactions, the rotorcraft have always presented a special challenge to system identification methods. A completely integrated methodology has been developed under this NASA contract to solve this difficult problem. This methodology has also been translated into a user oriented series of computer programs. This volume provides basic guidelines for efficient and effective use of one of these computer programs.

Figure 1 shows a schematic flowchart of the overall data processing technique for rotorcraft. The first step in this procedure is state estimation and instrument calibration. This is implemented by the computer program DEKFIS (for Discrete Extended Kalman Filter and Smoother) which implements an extended Kalman filter/smoother using the Friedland-Duffy formulation. Instrument biases and scale factors are estimated at this stage together with any state which is not measured directly. The second step involves estimation of the mathematical model of various forces, moments and interchanges. This is implemented in OSR (Optimal Subset Regression) computer program which uses a regression technique. Accurate estimates of parameters are obtained in the final step. One of two computer programs is used for this purpose. SCIDNT implements the maximum likelihood method for linear systems and NLSCIDNT extends the method to nonlinear rotorcraft models.

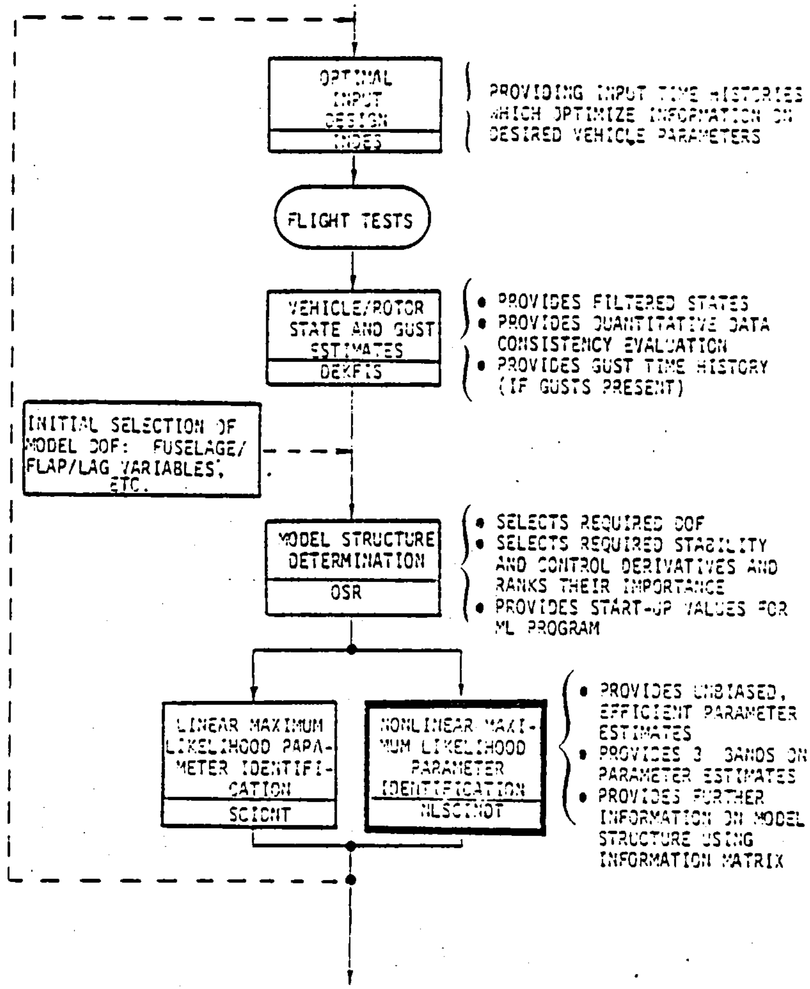


Figure 1 Integrated Rotorcraft System Identification Procedure

Accuracy of parameter estimates may be improved by using flight test inputs based on the input design program, INDES.

This user's manual describes the NLSCINDT computer program. The details of the theory and the particular implementation used are given in the final report.*

---

* Hall, W.E., Gupta, N.K., Hansen, R. and Bohn, J., "State Estimation and Parameter Identification for Rotorcraft," final report on contract NAS1-14549, May 1978.

This user's guide describes the structure of the NLSCIDNT program, the program's input and output, and illustrates them with example runs. It also gives guidelines for the program's effective use and information on selected topics which the reader will find helpful in using the program.

The contract research effort which led to the results in this report was financially supported by the Structures Laboratory, USARTL (AVRADCOM), NASA Langley Research Center and NASA Ames Research Center.

CHAPTER II
BACKGROUND

The nonlinear, maximum likelihood, parameter identification computer program (NLSCIDNT) described in this user's guide was written by Systems Control, Inc. (Vt) to evaluate rotorcraft stability and control coefficients from flight test data.

The program has the following features:

- The optimal estimates of the parameters (stability and control coefficients) are determined (identified) by minimizing the negative log likelihood cost function. These maximum likelihood estimates are asymptotically unbiased, consistent, and efficient (see Appendix A) [1,2].

- The minimization technique is the Levenberg-Marquardt method, which behaves like the steepest descent method when it is far from the minimum and behaves like the modified Newton-Raphson method when it is nearer the minimum. Hence, the technique becomes quadratically convergent as it nears the minimum, and at the same time it avoids the divergence problems often associated with quadratic techniques when far from the minimum.

- Twenty one states and 40 measurement variables are modeled, and any subset may be selected. States which are not integrated may be fixed at an input value, or time history data may be substituted for the state in the equations of motion.

- Any aerodynamic coefficient may be expressed as a nonlinear polynomial function of selected "expansion variables." This feature gives the user great flexibility to model nonlinear aerodynamics through the program's input without any changes in the program's code. Up to five expansion variables may be selected from a list of 17.

- The states and state sensitivities (partial derivatives of the states with respect to the identified parameters) are propagated by an efficient implementation of a variable-order, variable-stepsize, Adams-Bashforth predictor and Adams-Moulton corrector algorithm due to Shampine and Gordon [3].

CHAPTER III
PROGRAM DESCRIPTION

3.1 PROGRAM STRUCTURE

The overall logic of the NLSCIDNT program is presented in Fig. 3.1. It is intended to give the user a general overview of the program and, therefore, it shows only the major routines with a brief description of their purpose.

The subroutine calling structure is given in Fig. 3.2. The user may find this figure useful in further understanding the program's flow and in constructing an overlay structure if one is needed. The program was developed on a CDC 7600 and required 303,771 (octal), 100,345 (decimal) storage locations (words).

The functions of the various subprograms are sketeched below:

DRIVER is the main routine. It performs no calculations itself, but rather calls three routines--INPUT, SMAIN, and ØUTPUT--which accomplish the computational and input/output tasks. DRIVER's most important function is the dimensioning of the major arrays and setting up most of the labeled commons in the program.

BLØCKD is a block data routine. Its major function is to initialize arrays which contain labels for the measurement, control, and expansion variables for input/output purposes.

INPUT handles all of the program's input, checks for errors in the inputs, and initializes various flags within the program. It also prints this information as a record for the user.

FLAGIT is used by INPUT to set pointer arrays to the state, measurement, and expansion variables which are to be used in the run.

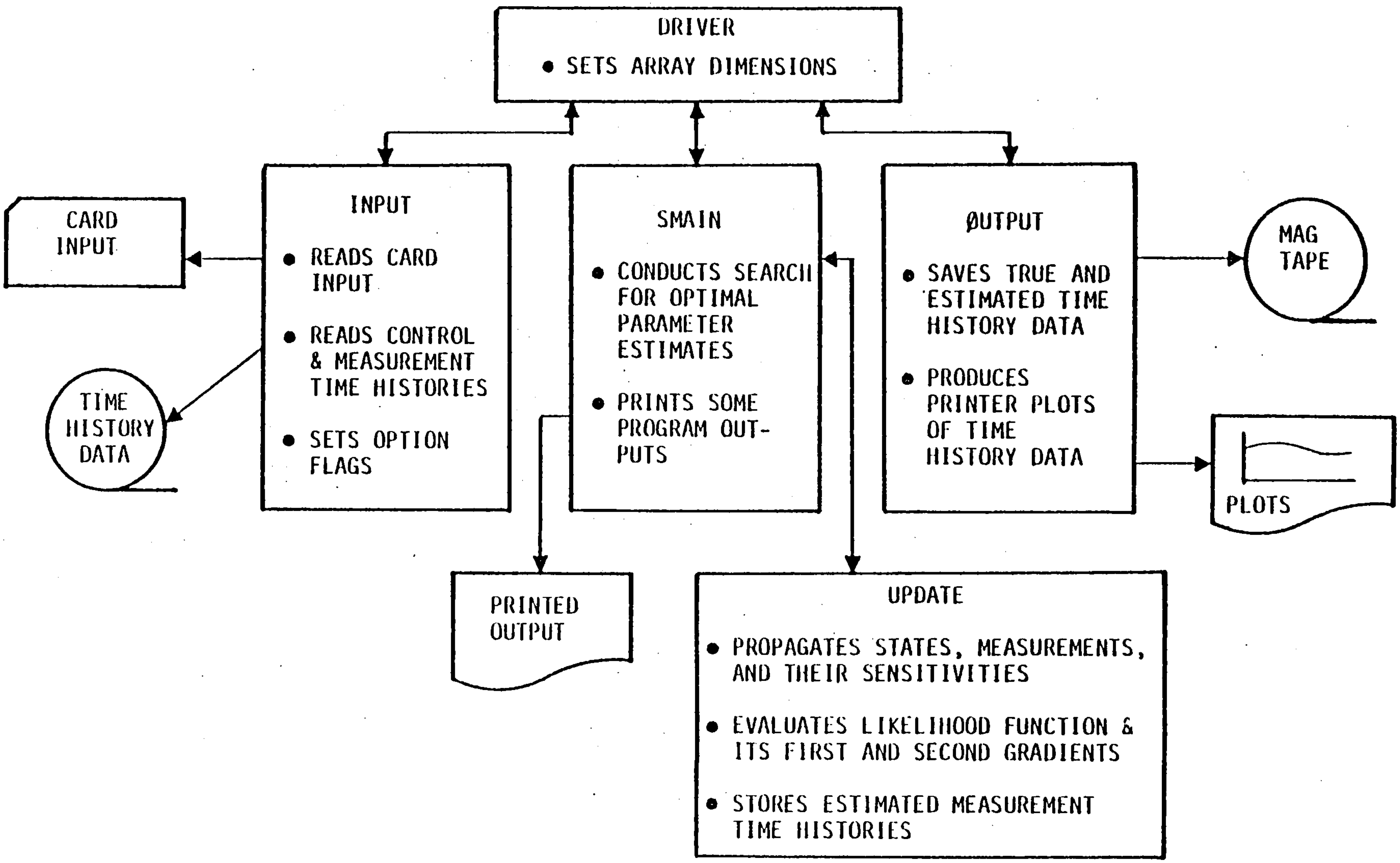
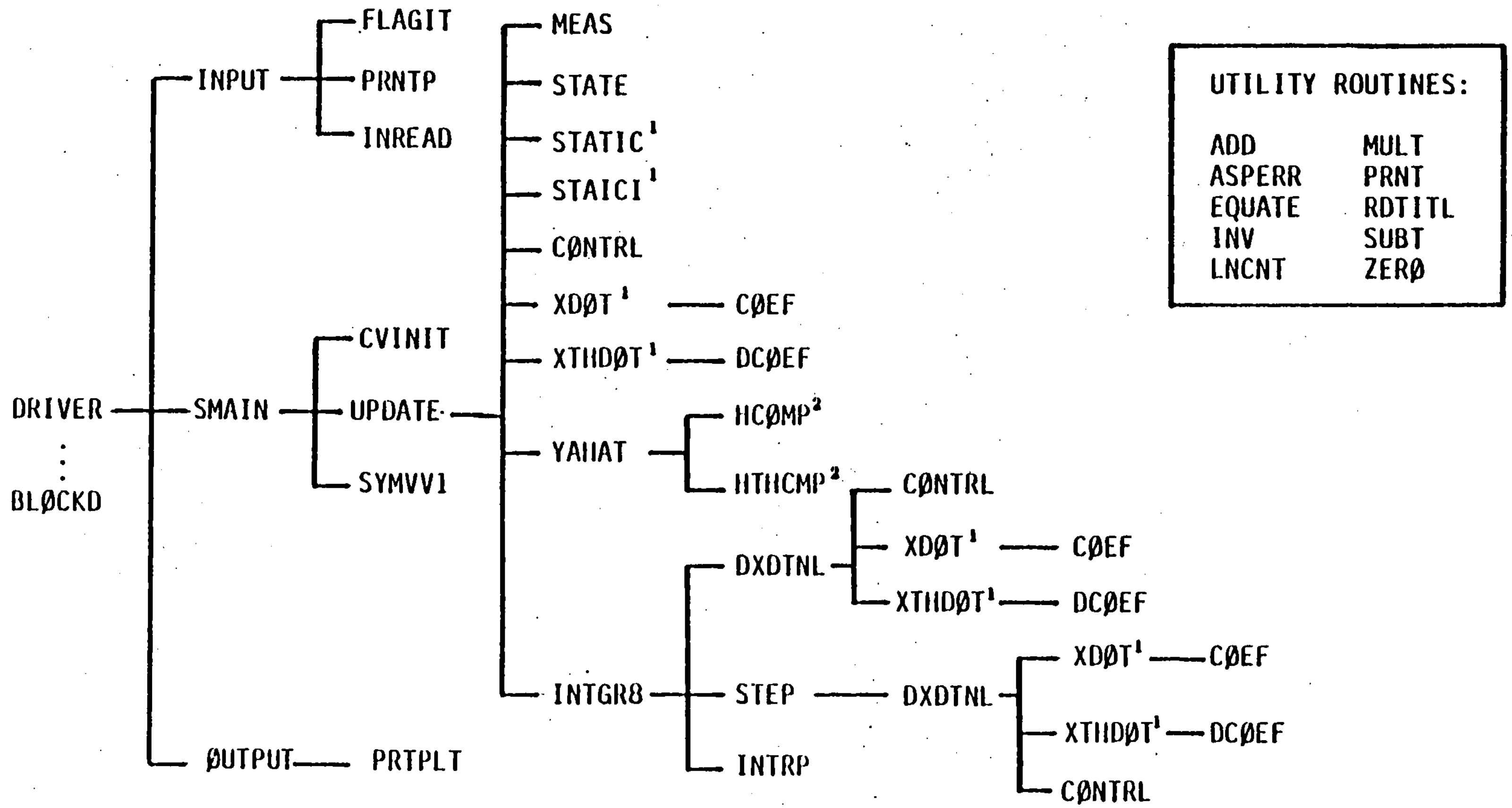


Figure 3.1 NLSCIDNT Functional Flow Chart



[1] Entry point in STATE
[2] Entry point in MEAS

Figure 3.2 Subroutine Calling Structure

INREAD is called by INPUT to read the time histories of the measurement and control variables.

PRNTP is called by INPUT to print the initial parameter values which were read into the program.

SMAIN performs the main computational tasks. It searches for the optimal parameter estimates by means of the Levenberg-Marquardt minimization technique. It constrains the parameter values within bounds set by the user, and it automatically restricts the search for estimates of parameters having low identifiability. This latter function is accomplished by adaptively modifying the effective rank of the information matrix.

CVINIT is called by SMAIN to compute the covariances of the measurements and controls.

SYMVV1 is called by SMAIN to compute the eigenvalues and eigenvectors of the (symmetric) information matrix. It is optimized for use on symmetric matrices. The eigensystem is calculated by the QR algorithm after the matrix has been put in tridiagonal form by a Householder transformation.

UPDATE: (1) propagates the state estimates ($\hat{x}$) and state sensitivities ($\partial\hat{x}/\partial\hat{\theta}$) with respect to the parameter estimates and computes the measurement estimates ($\hat{y}$) and their sensitivities ($\partial\hat{y}/\partial\hat{\theta}$) by calls to appropriate subroutines; (2) calculates the cost function and its first and second gradients for use by SMAIN; and (3) stores the measurement estimate time histories for use by ØUTPUT.

INTGR8 is called by UPDATE to oversee the integration of the first-order differential equations for $\hat{x}$ and $\partial\hat{x}/\partial\hat{\theta}$ forward one sample time increment. This is accomplished by a variable-order, variable-stepsize, predictor-corrector algorithm. The algorithm chooses its own order and stepsize to most efficiently satisfy relative and absolute error bounds on $\hat{x}$ and $\partial\hat{x}/\partial\hat{\theta}$ set by the user. The resulting stepsize may be more as well as less than the sample time increment.

STEP is called by INTGR8 to perform the actual integration of the equations. In general, the integration stepsize is not a rational fraction of the sample time interval.

INTRP is called by INTGR8 to interpolate the values of $\hat{x}$, $\partial\hat{x}/\partial\hat{\theta}$ and their time derivatives at the data sampling times.

DXDTNL is called usually by STEP but initially by INTGR8 to obtain the time derivatives of $\hat{x}$ and also $\partial\hat{x}/\partial\hat{\theta}_i$, where $\hat{\theta}_i$ is an identified parameter.

YAHAT is called by UPDATE to evaluate $\hat{y}$ and $\partial\hat{y}/\partial\hat{\theta}$ at the data sample times given $\hat{x}$ and $\partial\hat{x}/\partial\hat{\theta}$ at those times.

STATE has five ENTRY points. A call to STATE initializes variables used elsewhere in the subroutine. STATIC sets the state initial condition estimates, $\hat{x}(0)$. STAICI sets the state sensitivity initial condition estimates, $\partial\hat{x}(0)/\partial\hat{\theta}$. A call to XDØT computes the time derivative of the state estimates, $d\hat{x}/dt$. A call to XTHDØT computes the time derivative of the state sensitivities with respect to a specified parameter, $d(\partial\hat{x}/\partial\hat{\theta}_i)/dt$. Therefore, the vehicle equations of motion appear in this subroutine.

MEAS has three ENTRY points. A call to MEAS evaluates the measurement noise covariance matrix, R. HCØMP computes the measurement estimates, $\hat{y}$. HTHCMP computes the measurement sensitivities with respect to a specified parameter, $\partial\hat{y}/\partial\hat{\theta}_i$. Therefore, the measurement instrument equations appear in this routine.

CØEF evaluates the aerodynamic coefficients required by the vehicle equations of motion in subroutine STATE.

DCØEF evaluates the partial derivatives of the aerodynamic coefficients with respect to each of the identified parameters.

CØNTRL finds the values of the control inputs at any specified time by linear interpolation between the nearest sample time points.

ØUTPUT performs three tasks: (1) it computes the combined optimal parameters if the a priori information matrix is not zero (see Section 5.3); (2) it writes the time histories of the controls, measurements, and measurement estimates to tape; and (3) it uses PRTPLT to produce plots of the estimated measurement time history fit to the actual measurement time histories and to plot the control input time histories.

In addition to these major subroutines, there are ten utility routines of which use is made by several of the above routines.

ADD finds the sum of two matrices.

EQUATE sets one matrix equal to another.

INV finds the inverse and determinant of a matrix.

MULT finds the product of two matrices.

PRNT writes a matrix on the printer.

RDTITL reads a card containing the run title.

SUBT finds the difference of two matrices.

ZERØ sets a matrix equal to the zero matrix.

ASPERR produces a "walk-back" when an error in computation is detected.

LNCNT prints the run's title at the top of each page.

PRTPLT produces plots on the printer.

3.2 EQUATIONS OF MOTION

The maximum likelihood identification algorithm may be applied to any dynamic system. NLSCIDNT was coded specifically to model the motions of a rotorcraft. These equations are written in terms of the 23 state variables and eight controls listed in Table 3.1.

Table 3.1
State, Control, and Measurement Variables

STATE VARIABLES			
INDEX	SYMBOL	DEFINITION	UNITS
1	u	longitudinal component of velocity	ft/sec
2	v	lateral component of velocity	ft/sec
3	w	vertical component of velocity	ft/sec
4	p	body roll rate	rad/sec
5	q	body pitch rate	rad/sec
6	r	body yaw rate	rad/sec
7	ϕ	Euler roll angle	rad
8	θ	Euler pitch angle	rad
9	ψ	Euler yaw angle	rad
10	$\dot{\beta}_o$	collective flap angular rate	rad/sec
11	$\dot{\beta}_{1c}$	longitudinal flap angular rate	rad/sec
12	$\dot{\beta}_{1s}$	lateral flap angular rate	rad/sec
13	β_o	collective flap angle	rad
14	β_{1c}	longitudinal flap angle	rad
15	β_{1s}	lateral flap angle	rad
16	$\dot{\zeta}_o$	collective lag angular rate	rad/sec
17	$\dot{\zeta}_{1c}$	longitudinal lag angular rate	rad/sec
18	$\dot{\zeta}_{1s}$	lateral lag angular rate	rad/sec
19	$\delta_{\Omega r}$	rotor speed variation	rad/sec
20	ζ_o	collective lag angle	rad
21	ζ_{1c}	longitudinal lag angle	rad
22	ζ_{1s}	lateral lag angle	rad
23	$\delta_{\psi r}$	rotor angular position variation	rad

Table 3.1 (Continued)

CONTROL VARIABLES			
INDEX	SYMBOL	DEFINITION	UNITS
1	θ_o	collective pitch of blades	rad
2	θ_{1c}	lateral cyclic pitch of blades	rad
3	θ_{1s}	longitudinal cyclic pitch of blades	rad
4	δ_{TR}	pitch of tail rotor blades	rad
5	δ_e	elevator angle	rad
6	δ_a	aileron angle	rad
7	δ_r	rudder angle	rad
8	δ_f	flaperon angle	rad

MEASUREMENT VARIABLES			
INDEX	SYMBOL	DEFINITION	UNITS
1	a_x	longitudinal accelerometer	ft/sec^2
2	a_y	lateral accelerometer	ft/sec^2
3	a_z	vertical accelerometer	ft/sec^2
4	$\dot{p}_m$	roll angular accelerometer	rad/sec^2
5	$\dot{q}_m$	pitch angular accelerometer	rad/sec^2
6	$\dot{r}_m$	yaw angular accelerometer	rad/sec^2
7	p_m	roll rate gyro	rad/sec
8	q_m	pitch rate gyro	rad/sec
9	r_m	yaw rate gyro	rad/sec
10	ϕ_m	roll position gyro	rad
11	θ_m	pitch position gyro	rad
12	ψ_m	yaw position gyro	rad
13	α_m	angle-of-attack vane	rad
14	β_m	sideslip vane	rad
15	V_m	pitot tube	ft/sec

Table 3.1 (Continued)

MEASUREMENT VARIABLES (Cont'd)			
INDEX	SYMBOL	DEFINITION	UNITS
16	u_m	longitudinal velocity	ft/sec
17	v_m	lateral velocity	ft/sec
18	w_m	vertical velocity	ft/sec
19	X_{R_m}	rotor longitudinal force	lbs
20	Y_{R_m}	rotor lateral force	lbs
21	Z_{R_m}	rotor vertical force	lbs
22	L_{R_m}	rotor roll moment	ft-lbs
23	M_{R_m}	rotor pitch moment	ft-lbs
24	N_{R_m}	rotor yaw moment	ft-lbs
25	β_{1m}	flap angle of blade 1	rad
26	β_{2m}	flap angle of blade 2	rad
27	β_{3m}	flap angle of blade 3	rad
28	β_{4m}	flap angle of blade 4	rad
29	β_{5m}	flap angle of blade 5	rad
30	β_{6m}	flap angle of blade 6	rad
31	β_{7m}	flap angle of blade 7	rad
32	ζ_{1m}	lag angle of blade 1	rad
33	ζ_{2m}	lag angle of blade 2	rad
34	ζ_{3m}	lag angle of blade 3	rad
35	ζ_{4m}	lag angle of blade 4	rad
36	ζ_{5m}	lag angle of blade 5	rad
37	ζ_{6m}	lag angle of blade 6	rad
38	ζ_{7m}	lag angle of blade 7	rad
39	$(\cos \psi_R)_m$	cosine of rotor azimuth	N.D.
40	$(\sin \psi_R)_m$	sine of rotor azimuth	N.D.

The equations of motion are lengthy for the sake of completeness. As a result, they contain terms which will frequently be negligible, and they contain far more aerodynamic coefficients than can be accurately identified with ordinary rotorcraft instrumentation. With properly designed control inputs and with highly accurate and very complete instrumentation such that all state variables are redundantly observable, it would be possible to use the complete set of equations of motion and identify all significant parameters. However, in the common use of the program, only subsets of the equations are integrated and only the parameters appearing in those integrated equations are identified.

For this purpose, the user has the option of fixing any state at its initial value or reading in a time history for it rather than letting the program propagate that state by integrating its differential equation. A considerable savings in computation time is realized if states unnecessary to the problem at hand are not integrated. As an example of the use of these options, suppose lateral directional data about a trim condition is to be processed but significant perturbations in pitch, angle-of-attack and flapping occurred. Then the user may choose to have the lateral states v, p, r, and ϕ integrated; to hold u, the rotor lag and speed states at their initial values, and to substitute measurements of q, θ, w (perhaps derived from angle-of-attack) and the six flapping states for these state variables in the equations of motion.

The 13 degree-of-freedom, nonlinear equations of motion are in body-axis coordinates with origin at the center of gravity.

$$W\dot{x} = f \tag{3.1}$$

The state vector is

$$x = [u \; v \; w \; p \; q \; r \; \phi \; \theta \; \psi \;\; \dot{\beta}_o \; \dot{\beta}_{1c} \; \dot{\beta}_{1s} \; \beta_o \; \beta_{1c} \; \beta_{1s}$$
$$\dot{\zeta}_o \; \dot{\zeta}_{1c} \; \dot{\zeta}_{1s} \; \delta\Omega_R \; \zeta_o \; \zeta_{1c} \; \zeta_{1s} \; \delta\psi_R]^T \tag{3.2}$$

The vector f is composed of the following elements:

$$f_u = vr - w_o q - g \sin\theta + \frac{1}{m} \rho\Omega^2 R^4 \pi \{ C_x + T_{aero} \sin\theta_R - H_{aero} \cos\theta_R \} \qquad (3.3)$$

$$f_v = wp - ur + g \cos\theta \sin\phi + \frac{1}{m} \rho\Omega^2 R^4 \pi \{ C_y + Y_{aero} \} \qquad (3.4)$$

$$f_w = uq - vp + g \cos\theta \cos\phi + \frac{1}{m} \rho\Omega^2 R^4 \pi \{ C_z - T_{aero} \cos\theta_R - H_{aero} \sin\theta_R \} \qquad (3.5)$$

$$\begin{aligned} f_p = {} & (I_y - I_z)qr + I_{yz}(q^2 - r^2) + (I_{xz}q - I_{xy}r)p \\ & + \rho\Omega^2 R^5 \pi \{ C_L + Q_{aero} \sin\theta_R - L_{aero} \cos\theta_R \\ & - Y_{aero} z_{hub} + N_b \cos\theta_R (I_o q' - I_{\dot\beta\alpha} \beta'_{1c} - I_{\dot\zeta\alpha} \zeta'_{1c}) \} \end{aligned} \qquad (3.6)$$

$$\begin{aligned} f_q = {} & (I_z - I_x)pr + I_{xz}(r^2 - p^2) + (I_{xy}r - I_{yz}q)p \\ & + \rho\Omega^2 R^5 \pi \{ C_M + M_{aero} + H_{aero}(x_{hub} \sin\theta_R - z_{hub} \cos\theta_R) \\ & + T_{aero}(x_{hub} \cos\theta_R + z_{hub} \sin\theta_R) + N_b [I_{\dot\beta\alpha} \beta'_{1s} \\ & + I_{\dot\zeta\alpha} \zeta'_{1s} - I_o(p' \cos\theta_R + r' \sin\theta_R)] \} \end{aligned} \qquad (3.7)$$

$$\begin{aligned} f_r = {} & (I_x - I_y)pq + I_{xy}(p^2 - q^2) + (I_{yz}p - I_{xz}q)r \\ & + \rho\Omega^2 R^5 \pi \{ C_N + Y_{aero} x_{hub} - L_{aero} \sin\theta_R \\ & - Q_{aero} \cos\theta_R + N_b \sin\theta_R (-I_{\dot\beta\alpha} \beta'_{1c} - I_{\dot\zeta\alpha} \zeta'_{1c} + I_o q') \} \end{aligned} \qquad (3.8)$$

$$f_\phi = p + (q \sin\phi + r \cos\phi) \tan\theta \qquad (3.9)$$

$$f_\theta = q\cos\phi - r\sin\phi \qquad (3.10)$$

$$f_\psi = (q\sin\phi + r\cos\phi)/\cos\theta \qquad (3.11)$$

$$f_{\dot\beta_o} = -2I_{\beta\dot\psi}(p\sin\theta_R - r\cos\theta_R) - I_\beta \nu_\beta^2 \beta_o$$

$$+ \rho\Omega^2 R^5\pi M^o \qquad (3.12)$$

$$f_{\dot\beta_{1c}} = -I_\beta(\nu_\beta^2 - 1)\beta_{1c} - (I_\beta g_s \nu_\beta + 2I_{\beta\dot\beta})\beta_{1s} \qquad (3.13)$$

$$+ 2I_{\beta\alpha}(p\cos\theta_R + r\sin\theta_R) + \rho\Omega^2 R^5\pi M^{1c} \qquad (3.13)$$

$$f_{\dot\beta_{1s}} = (I_\beta g_s \nu_\beta + 2I_{\beta\dot\beta})\beta_{1c} - I_\beta(\nu_\beta^2-1)\beta_{1s} - 2I_{\beta\alpha}q$$

$$+ \rho\Omega^2 R^5\pi M^{1s} \qquad (3.14)$$

$$f_{\beta_o} = \dot\beta_o \qquad (3.15)$$

$$f_{\beta_{1c}} = \dot\beta_{1c} \qquad (3.16)$$

$$f_{\beta_{1s}} = \dot\beta_{1s} \qquad (3.17)$$

$$f_{\dot\zeta_o} = -2I_{\zeta\dot\psi}(p\sin\theta_R - r\cos\theta_R) + \rho\Omega^2 R^5\pi Q^o \qquad (3.18)$$

$$f_{\dot\zeta_{1c}} = -I_\zeta(\nu_\zeta^2-1)\zeta_{1c} - (I_\zeta g_s \nu_\zeta + 2I_{\zeta\dot\zeta})\zeta_{1s} + \rho\Omega^2 R^5\pi Q^{1c} \qquad (3.19)$$

$$f_{\dot\zeta_{1s}} = (I_\zeta g_s \nu_\zeta + 2I_{\zeta\dot\zeta})\zeta_{1c} - I_\zeta(\nu_\zeta^2-1)\zeta_{1s}$$

$$+ \rho\Omega^2 R^2\pi Q^{1s} \qquad (3.20)$$

$$f_{\delta\Omega_R} = \rho\Omega^2 R^5\pi Q^\Omega \qquad (3.21)$$

$$f_{\zeta_o} = \dot\zeta_o \qquad (3.22)$$

$$f_{\zeta_{1c}} = \dot{\zeta}_{1c} \tag{3.23}$$

$$f_{\zeta_{1s}} = \dot{\zeta}_{1s} \tag{3.24}$$

$$f_{\delta\Omega_R} = \delta\dot{\Omega}_R \tag{3.25}$$

where

$$C_x = C_{x_o} + C_{x_u}\frac{u}{\Omega R} + C_{x_v}\frac{v}{\Omega R} + C_{x_w}\frac{w}{\Omega R} + C_{x_p}\frac{p}{\Omega}$$

$$+ C_{x_q}\frac{q}{\Omega} + C_{x_r}\frac{r}{\Omega} + C_{x\delta_{TR}}\delta_{TR} + C_{x_{\delta_e}}\delta_e$$

$$+ C_{x\delta_a}\delta_a + C_{x\delta_r}\delta_r + C_{x\delta_f}\delta_f \tag{3.26}$$

and C_Y, C_Z, C_L, C_M and C_N are expanded similarly;

$$T_{aero} = T_o + T_{\dot{\beta}_o}\frac{\dot{\beta}_o}{\Omega} + T_{\dot{\beta}_{1c}}\frac{\dot{\beta}_{1c}}{\Omega} + T_{\dot{\beta}_{1s}}\frac{\dot{\beta}_{1s}}{\Omega} + T_{\beta_o}\beta_o$$

$$+ T_{\beta_{1c}}\beta_{1c} + T_{\beta_{1s}}\beta_{1s} + T_{\dot{\zeta}_o}\frac{\dot{\zeta}_o}{\Omega} + T_{\dot{\zeta}_{1c}}\frac{\dot{\zeta}_{1c}}{\Omega}$$

$$+ T_{\dot{\zeta}_{1s}}\frac{\dot{\zeta}_{1s}}{\Omega} + T_{\delta\Omega_R}\frac{\delta\Omega_R}{\Omega} + T_{\zeta_o}\zeta_o + T_{\zeta_{1c}}\zeta_{1c}$$

$$+ T_{\zeta_{1s}}\zeta_{1s} + T_{\delta\psi_R}\delta\psi_R + T_{\theta_o}\theta_o + T_{\theta_{1c}}\theta_{1c} + T_{\theta_{1s}}\theta_{1s} \tag{3.27}$$

and H_{aero}, Y_{aero}, L_{aero}, M_{aero}, and Q_{aero} are expanded similarly;

$$M^{o} = M^{o}_{o} + M^{o}_{u}\frac{u}{\Omega R} + M^{o}_{v}\frac{v}{\Omega R} + M^{o}_{w}\frac{w}{\Omega R} + M^{o}_{p}\frac{p}{\Omega} + M^{o}_{q}\frac{q}{\Omega} + M^{o}_{r}\frac{r}{\Omega}$$

$$+ M^{o}_{\dot{\beta}_o}\frac{\dot{\beta}_o}{\Omega} + M^{o}_{\dot{\beta}_{1c}}\frac{\dot{\beta}_{1c}}{\Omega} + M^{o}_{\dot{\beta}_{1s}}\frac{\dot{\beta}_{1s}}{\Omega} + M^{o}_{\beta_o}\beta_o + M^{o}_{\beta_{1c}}\beta_{1c}$$

$$+ M^{o}_{\beta_{1s}}\beta_{1s} + M^{o}_{\dot{\zeta}_o}\frac{\dot{\zeta}_o}{\Omega} + M^{o}_{\dot{\zeta}_{1c}}\frac{\dot{\zeta}_{1c}}{\Omega} + M^{o}_{\dot{\zeta}_{1s}}\dot{\zeta}_{1s} + M^{o}_{\delta\Omega_R}\frac{\delta\Omega_R}{\Omega}$$

$$+ M^{o}_{\zeta_o}\zeta_o + M^{o}_{\zeta_{1c}}\zeta_{1c} + M^{o}_{\zeta_{1s}}\zeta_{1s} + M^{o}_{\delta\psi_R}\delta\psi_R + M^{o}_{\theta_o}\theta_o$$

$$+ M^{o}_{\theta_{1c}}\theta_{1c} + M^{o}_{\theta_{1s}}\theta_{1s} \qquad (3.28)$$

and M^{1c}, M^{1s}, Q^{o}, Q^{1c}, Q^{1s}, and Q^{Ω} are expanded similarly.

The elements of the matrix W which are not zero are:

$$W_{uu} = 1$$

$$W_{u\dot{\zeta}_{1s}} = \frac{\rho\Omega^2R^4\pi}{m} \cdot N_b \cdot \frac{S_\beta}{\Omega^2}\sin\theta_R$$

$$W_{u\dot{\beta}_o} = \frac{\rho\Omega^2R^4\pi}{m}\,\frac{N_b}{2}\,\frac{S_\zeta}{\Omega^2}\cos\theta_R$$

$$W_{vv} = 1$$

$$W_{v\dot{\zeta}_{1c}} = \frac{\rho\Omega^2R^4\pi}{m} \cdot \frac{N_b}{2} \cdot \frac{S_\zeta}{\Omega^2}$$

$$W_{ww} = 1$$

$$W_{w\dot{\beta}_o} = \frac{\rho\Omega^2R^4\pi}{m} \cdot N_b \cdot \frac{S_\beta}{\Omega^2} \cos\theta_R$$

$$W_{w\dot{\zeta}_{1s}} = \frac{\rho\Omega^2R^4\pi}{m} \frac{N_b}{2} \frac{S_\zeta}{\Omega^2} \sin\theta_R$$

$$W_{pp} = I_x$$

$$W_{pq} = -I_{xy}$$

$$W_{pr} = -I_{xz}$$

$$W_{p\dot{\beta}_{1s}} = -\rho\Omega^2R^5\pi \frac{N_b}{2} \cdot \frac{I_{\beta\alpha}}{\Omega^2} \cdot \cos\theta_R$$

$$W_{p\beta_o} = \rho\Omega^2R^5\pi \cdot (2N_b) \frac{I_{\dot{\beta}\psi}}{\Omega} \cdot \sin\theta_R$$

$$W_{p\beta_{1c}} = \rho\Omega^2R^5\pi N_b \frac{I_{\beta\alpha}}{\Omega} \cos\theta_R$$

$$W_{p\beta_{1s}} = -\rho\Omega^2R^5\pi N_b \frac{I_{\dot{\beta}\alpha}}{\Omega} \cos\theta_R$$

$$W_{p\dot{\zeta}_o} = -\rho\Omega^2R^5\pi N_b \frac{I_{\zeta\alpha}}{\Omega} \sin\theta_R$$

$$W_{p\dot{\zeta}_{1c}} = -\rho\Omega^2R^5\pi \frac{N_b}{2} \frac{S_\zeta}{\Omega^2} \frac{Z_{hub}}{R}$$

$$W_{p\dot{\psi}_s} = -\rho\Omega^2R^5\pi \frac{I_o}{\Omega^2}$$

$$W_{p\zeta_o} = \rho\Omega^2R^5\pi \; (2N_b) \; \frac{I_{\dot{\zeta}\psi}}{\Omega} \sin\theta_R$$

$$W_{p\zeta_{1s}} = -\rho\Omega^2R^5\pi \cdot N_b \; \frac{I_{\dot{\zeta}\alpha}}{\Omega} \cos\theta_R$$

$$W_{qp} = -I_{xy}$$

$$W_{qq} = I_y$$

$$W_{qr} = -I_{yz}$$

$$W_{q\dot{\beta}_o} = \rho\Omega^2R^5\pi \; N_b \; \frac{S_\beta}{\Omega^2} \; \frac{Z_{hub}}{R} \sin\theta_R + \frac{X_{hub}}{R} \cos\theta_R$$

$$W_{q\dot{\beta}_{1c}} = -\rho\Omega^2R^5\pi \; \frac{N_b}{2} \; \frac{I_{\beta\alpha}}{\Omega^2}$$

$$W_{q\beta_{1c}} = -\rho\Omega^2R^5\pi \; N_b \; \frac{I_{\dot{\beta}\alpha}}{\Omega}$$

$$W_{q\beta_{1s}} = -\rho\Omega^2R^5\pi \; N_b \; \frac{I_{\beta\alpha}}{\Omega}$$

$$W_{q\dot{\zeta}_{1s}} = \rho\Omega^2R^5\pi \; \frac{N_b}{2} \; \frac{S_\zeta}{\Omega^2} \left(\frac{Z_{hub}}{R} \cos\theta_R - \frac{X_{hub}}{R} \sin\theta_R \right)$$

$$W_{q\zeta_{1c}} = -\rho\Omega^2R^5\pi \; N_b \; \frac{I_{\dot{\zeta}\alpha}}{\Omega}$$

$$W_{rp} = -I_{xz}$$

$$W_{rq} = -I_{yz}$$

$$W_{rr} = I_z$$

$$W_{r\dot{\beta}_{1s}} = -\rho\Omega^2R^5\pi \frac{N_b}{2} \frac{I_{\beta\alpha}}{\Omega^2} \sin \theta_R$$

$$W_{r\beta_o} = -\rho\Omega^2R^5\pi (2N_b) \frac{I_{\dot{\beta}\psi}}{\Omega} \cos \theta_R$$

$$W_{r\beta_{1c}} = \rho\Omega^2R^5\pi N_b \frac{I_{\beta\alpha}}{\Omega} \sin \theta_R$$

$$W_{r\beta_{1s}} = -\rho\Omega^2R^5\pi N_b \frac{I_{\beta\alpha}}{\Omega} \sin \theta_R$$

$$W_{r\dot{\zeta}_o} = \rho\Omega^2R^5\pi N_b \frac{I_{\zeta\alpha}}{\Omega^2} \cos \theta_R$$

$$W_{r\dot{\zeta}_{1c}} = \rho\Omega^2R^5\pi \frac{N_b}{2} \frac{S_\zeta}{\Omega^2} \frac{X_{hub}}{R}$$

$$W_{r\dot{\psi}_s} = \rho\Omega^2R^5\pi N_b \frac{I_o}{\Omega^2} \cos \theta_R$$

$$W_{r\zeta_o} = -\rho\Omega^2R^5\pi (2N_b) \frac{I_{\dot{\zeta}\psi}}{\Omega} \cos \theta_R$$

$$W_{r\zeta_{1s}} = -\rho\Omega^2R^5\pi N_b \frac{I_{\dot{\zeta}\alpha}}{\Omega} \sin \theta_R$$

$$W_{\phi\phi} = 1$$

$$W_{\theta\theta} = 1$$

$$W_{\psi\psi} = 1$$

$$W_{\dot{\beta}_o u} = S_\beta \sin \theta_R$$

$$W_{\dot{\beta}_o w} = -S_\beta \cos \theta_R$$

$$W_{\dot{\beta}_o q} = - S_\beta (z_{hub} \sin \theta_R - x_{hub} \cos \theta_R)$$

$$W_{\dot{\beta}_o \dot{\beta}_o} = I_\beta$$

$$W_{\dot{\beta}_o \beta_o} = I_\beta \, g_S \, \nu_\beta + 2I_{\beta\dot{\beta}}$$

$$W_{\dot{\beta}_o \psi_s} = 2I_{\beta\dot{\psi}}$$

$$W_{\dot{\beta}_{1c} q} = -I_{\beta\alpha}$$

$$W_{\dot{\beta}_{1c} \dot{\beta}_{1c}} = I_\beta$$

$$W_{\dot{\beta}_{1c} \beta_{1c}} = I_\beta \, g_S \, \nu_\beta + 2I_{\beta\dot{\beta}}$$

$$W_{\dot{\beta}_{1c} \beta_{1s}} = 2I_\beta$$

$$W_{\dot{\beta}_{1s} p} = -I_{\beta\alpha} \cos \theta_R$$

$$W_{\dot{\beta}_{1s} r} = -I_{\beta\alpha} \sin \theta_R$$

$$W_{\dot{\beta}_{1s}\dot{\beta}_{1s}} = I_\beta$$

$$W_{\dot{\beta}_{1s}\beta_{1c}} = -2I_\beta$$

$$W_{\dot{\beta}_{1s}\beta_{1s}} = I_\beta \ g_S \ \nu_\beta + 2I_{\beta\dot{\beta}}$$

$$W_{\beta_o\beta_o} = 1$$

$$W_{\beta_{1c}\beta_{1c}} = 1$$

$$W_{\beta_{1s}\beta_{1s}} = 1$$

$$W_{\dot{\zeta}_o p} = I_{\zeta\alpha} \sin \theta_R$$

$$W_{\dot{\zeta}_o r} = -I_{\zeta\alpha} \cos \theta_R$$

$$W_{\dot{\zeta}_o\dot{\zeta}_o} = I_\zeta$$

$$W_{\dot{\zeta}_o\dot{\psi}_s} = I_{\zeta\alpha}$$

$$W_{\dot{\zeta}_o\zeta_o} = I_\zeta \ g_S \ \nu_\zeta + 2I_{\zeta\dot{\zeta}}$$

$$W_{\dot{\zeta}_o\psi_S} = 2I_{\zeta\dot{\psi}}$$

$$W_{\dot{\zeta}_{1c}v} = S_\zeta$$

$$W_{\dot{\zeta}_{1c}p} = -S_\zeta \, z_{hub}$$

$$W_{\dot{\zeta}_{1c}r} = S_\zeta \, x_{hub}$$

$$W_{\dot{\zeta}_{1c}\dot{\zeta}_{1c}} = I_\zeta$$

$$W_{\dot{\zeta}_{1c}\zeta_{1c}} = I_\zeta \, g_S \, \nu_\zeta + 2I_{\zeta\dot{\zeta}}$$

$$W_{\dot{\zeta}_{1c}\zeta_{1s}} = 2I_\zeta$$

$$W_{\dot{\zeta}_{1s}u} = S_\zeta \cos \theta_R$$

$$W_{\dot{\zeta}_{1s}w} = S_\zeta \sin \theta_R$$

$$W_{\dot{\zeta}_{1s}q} = S_\zeta(z_{hub} \cos \theta_R - x_{hub} \sin \theta_R)$$

$$W_{\dot{\zeta}_{1s}\dot{\zeta}_{1s}} = I_\zeta$$

$$W_{\dot{\zeta}_{1s}\zeta_{1c}} = -2I_\zeta$$

$$W_{\dot{\zeta}_{1s}\zeta_{1s}} = I_\zeta \, g_S \, \nu_\zeta + 2I_{\zeta\dot{\zeta}}$$

$$W_{\dot{\psi}_S\dot{\psi}_S} = 1$$

$$W_{\zeta_o\zeta_o} = 1$$

$W_{\zeta_{1c}\zeta_{1c}} = 1$

$W_{\zeta_{1s}\zeta_{1s}} = 1$

$W_{\psi_s\psi_s} = 1$

where

g = acceleration of gravity, ft/sec^2

ρ = density of the air, slug/ft^3

m = mass of the rotorcraft, slug

Ω = reference rotor speed, rad/sec

R = rotor radius, ft

I_x = roll moment of inertia = $\int(y^2 + z^2)dm$, slug-ft^2

I_y = pitch moment of inertia = $\int(x^2 + z^2)dm$, slug-ft^2

I_z = yaw moment of inertia = $\int(x^2 + y^2)dm$, slug-ft^2

I_{xy} = x-y product of inertia = $\int xy\, dm$, slug-ft^2

I_{xz} = x-z product of inertia = $\int xz\, dm$, slug-ft^2

I_{yz} = y-z product of inertia = $\int yz\, dm$, slug-ft^2

$$\left.\begin{aligned} S\beta &= \int_0^1 N_\beta\, m_b dr \\ S\zeta &= \int_0^1 N_\zeta\, m_b dr \end{aligned}\right\} \quad \text{All terms non-dimensional}$$

where

N_β = 1st flapping mode shape, N.D.

N_ζ = 1st lagging mode shape, N.D.

m_b = blade mass

´ implies division by ΩR. e.g. $p' = p/\Omega R$

3.3 MEASUREMENT EQUATIONS

Forty measurement instruments are modeled in the program, from which the user may select any subset. Different measurement equations are applicable depending upon the set of equations of motion selected.

3.3.1 Measurements for Complete Set

The equations which evaluate the measurement estimates when the complete set of equations of motion is used are:

(1) longitudinal accelerometer

$$a_x = \dot{u} - vr + wq + g \sin \theta - (q^2 + r^2)x_{cg_x} + (pq - \dot{r})y_{cg_x} + (pr + \dot{q})z_{cg_x} + b_x$$

(2) lateral accelerometer

$$a_y = \dot{v} - wp + ur - g \sin \phi \cos \theta + (pq + \dot{r})x_{cg_y} - (r^2 + p^2)y_{cg_y} + (rq - \dot{p})z_{cg_y} + b_y$$

(3) vertical accelerometer

$$az = \dot{w} - uq + vp - g \cos \phi \cos \theta + (pr - \dot{q})x_{cg_z} + (rq + \dot{p})y_{cg_z} - (q^2 + p^2)z_{cg_z} + b_z$$

(4) roll angular accelerometer

$$\dot{p}_m = \dot{p} + b_{\dot{p}}$$

(5) pitch angular accelerometer

$$\dot{q}_m = \dot{q} + b_{\dot{q}}$$

(6) yaw angular accelerometer

$$\dot{r}_m = \dot{r} + b_{\dot{r}}$$

(7) roll rate

$$p_m = p + b_p$$

(8) pitch rate

$$q_m = q + b_q$$

(9) yaw rate

$$r_m = r + b_r$$

(10) roll angle

$$\phi_m = \phi + b_\phi$$

(11) pitch angle

$$\theta_m = \theta + b_\theta$$

(12) yaw angle

$$\psi_m = \psi + b_\psi$$

(13) angle-of-attack vane

$$\alpha_m = (1 + k_\alpha) \tan^{-1}\left[\frac{w - qx_{cg_\alpha} + py_{cg_\alpha}}{u - ry_{cg_\alpha} + qz_{cg_\alpha}}\right] + b_\alpha$$

(14) angle-of-sideslip vane

$$\beta_m = (1 + k_\beta) \sin^{-1}\left[\frac{v + rx_{cg_\beta} - pz_{cg_\beta}}{V_T}\right] + b_\beta$$

where

$$V_T = \left\{\left[u - ry_{cg_\beta} + qz_{cg_\beta}\right]^2 + \left[v + rx_{cg_\beta} - pz_{cg_\beta}\right]^2 + \left[w - qx_{cg_\beta} + py_{cg_\beta}\right]^2\right\}^{1/2}$$

(15) total velocity

$$V_m = (1 + K_V) \sqrt{u^2 + v^2 + w^2} + b_V$$

(16) $u_m = (1 + k_u)u + b_u$

(17) $v_m = (1 + k_v)v + b_v$

(18) $w_m = (1 + k_w)w + b_w$

(19) $X_{R_m} = (1 + k_{X_R})\pi\rho\Omega^2R^4(T_{aero} \sin \theta_R - H_{aero} \cos \theta_R) + b_{X_R}$

(20) $Y_{R_m} = (1 + k_{Y_R})\pi\rho\Omega^2R^4\, Y_{aero} + b_{Y_R}$

(21) $Z_{R_m} = (1 + k_{Z_R})\pi\rho\Omega^2R^4(-H_{aero} \sin \theta_R - T_{aero} \cos \theta_R) + b_{Z_R}$

$$(22)\quad L_{R_m} = (1+k_{L_R})\pi\rho\Omega^2R^5(Q_{aero}\sin\theta_R - L_{aero}\cos\theta_R - Y_{aero}\tilde{z}_{ref}) + b_{L_R}$$

$$(23)\quad M_{R_m} = (1+k_{M_R})\pi\rho\Omega^2R^5[M_{aero} + H_{aero}(\tilde{x}_{ref}\sin\theta_R - \tilde{z}_{ref}\cos\theta_R) + T_{aero}(\tilde{x}_{ref}\cos\theta_R + \tilde{z}_{ref}\sin\theta_R)] + b_{M_R}$$

$$(24)\quad N_{R_m} = (1+k_{N_R})\pi\rho\Omega^2R^5(\tilde{x}_{ref}Y_{aero} - L_{aero}\sin\theta_R - Q_{aero}\cos\theta_R) + b_{N_R}$$

$$(25)\quad \beta_{1m} = (1+k_{\beta_1})[\beta_o+\beta_{o_{ref}} - (\beta_{1c}+\beta_{1c_{ref}})\cos(\Omega t+\theta_1) - (\beta_{1s}+\beta_{1s_{ref}})\sin(\Omega t+\theta_1)] + b_{\beta_1}$$

$$(26)\quad \beta_{2m} = (1+k_{\beta_2})[\beta_o+\beta_{o_{ref}} - (\beta_{1c}+\beta_{1c_{ref}})\cos(\Omega t+\theta_1+\Delta\theta_1) - (\beta_{1s}+\beta_{1s_{ref}})\sin(\Omega t+\theta_1+\Delta\theta_1)] + b_{\beta_2}$$

$$(27)\quad \beta_{3_m} = (1+k_{\beta_3})[\beta_o+\beta_{o_{ref}} - (\beta_{1c}+\beta_{1c_{ref}})\cos(\Omega t+\theta_1+2\Delta\theta_1) - (\beta_{1s}+\beta_{1s_{ref}})\sin(\Omega t+\theta_1+2\Delta\theta_1)] + b_{\beta_3}$$

$$(28)\quad \beta_{4_m} = (1+k_{\beta_4})[\beta_o+\beta_{o_{ref}} - (\beta_{1c}+\beta_{1c_{ref}})\cos(\Omega t+\theta_1+3\Delta\theta_1) - (\beta_{1s}+\beta_{1s_{ref}})\sin(\Omega t+\theta_1+3\Delta\theta_1)] + b_{\beta_4}$$

$$(29)\quad \beta_{5_m} = (1+k_{\beta_5})[\beta_o+\beta_{o_{ref}}-(\beta_{1c}+\beta_{1c_{ref}})\cos(\Omega t+\theta_1+4\Delta\theta_1)$$
$$-(\beta_{1s}+\beta_{1s_{ref}})\sin(\Omega t+\theta_1+4\Delta\theta_1)] + b_{\beta_5}$$

$$(30)\quad \beta_{6_m} = (1+k_{\beta_6})[\beta_o+\beta_{o_{ref}}-(\beta_{1c}+\beta_{1c_{ref}})\cos(\Omega t+\theta_1+5\Delta\theta_1)$$
$$-(\beta_{1s}+\beta_{1s_{ref}})\sin(\Omega t+\theta_1+5\Delta\theta_1)] + b_{\beta_6}$$

$$(31)\quad \beta_{7_m} = (1+k_{\beta_7})[\beta_o+\beta_{o_{ref}}-(\beta_{1c}+\beta_{1c_{ref}})\cos(\Omega t+\theta_1+6\Delta\theta_1)$$
$$-(\beta_{1s}+\beta_{1s_{ref}})\sin(\Omega t+\theta_1+6\Delta\theta_1)] + b_{\beta_7}$$

$$(32)\quad \zeta_{1_m} = (1+k_{\zeta_1})[\zeta_o+\zeta_{o_{ref}}-(\zeta_{1c}+\zeta_{1c_{ref}})\cos(\Omega t+\theta_1)$$
$$-(\zeta_{1s}+\zeta_{1s_{ref}})\sin(\Omega t+\theta_1)] + b_{\zeta_1}$$

$$(33)\quad \zeta_{2_m} = (1+k_{\zeta_2})[\zeta_o+\zeta_{o_{ref}}-(\zeta_{1c}+\zeta_{1c_{ref}})\cos(\Omega t+\theta_1+\Delta\theta_1)$$
$$-(\zeta_{1s}+\zeta_{1s_{ref}})\sin(\Omega t+\theta_1+\Delta\theta_1)] + b_{\zeta_2}$$

$$(34)\quad \zeta_{3_m} = (1+k_{\zeta_3})[\zeta_o+\zeta_{o_{ref}}-(\zeta_{1c}+\zeta_{1c_{ref}})\cos(\Omega t+\theta_1+2\Delta\theta_1)$$
$$-(\zeta_{1s}+\zeta_{1s_{ref}})\sin(\Omega t+\theta_1+2\Delta\theta_1)] + b_{\zeta_3}$$

$$(35)\quad \zeta_{4_m} = (1+k_{\zeta_4})[\zeta_o+\zeta_{o_{ref}}-(\zeta_{1c}+\zeta_{1c_{ref}})\cos(\Omega t+\theta_1+3\Delta\theta_1)$$
$$-(\zeta_{1s}+\zeta_{1s_{ref}})\sin(\Omega t+\theta_1+3\Delta\theta_1)] + b_{\zeta_4}$$

$$(36)\quad \zeta_{5_m} = (1+k_{\zeta_5})[\zeta_o+\zeta_{o_{ref}}-(\zeta_{1c}+\zeta_{1c_{ref}})\cos(\Omega t+\theta_1+4\Delta\theta_1)$$
$$-(\zeta_{1s}+\zeta_{1s_{ref}})\sin(\Omega t+\theta_1+4\Delta\theta_1)] + b_{\zeta_5}$$

$$(37)\quad \zeta_{6_m} = (1+k_{\zeta_6})[\zeta_o+\zeta_{o_{ref}}-(\zeta_{1c}+\zeta_{1c_{ref}})\cos(\Omega t+\zeta_1+5\Delta\theta_1)$$

$$-(\zeta_{1s}+\zeta_{1s_{ref}})\sin(\Omega t+\theta_1+5\Delta\theta_1)] + b_{\zeta_6}$$

$$(38)\quad \zeta_{7_m} = (1+k_{\zeta_7})[\zeta_o+\zeta_{o_{ref}}-(\zeta_{1c}+\zeta_{1c_{ref}})\cos(\Omega t+\theta_1+6\Delta\theta_1)$$

$$-(\zeta_{1s}+\zeta_{1s_{ref}})\sin(\Omega t+\theta_1+6\Delta\theta_1)] + b_{\zeta_7}$$

$$(39)\quad (\sin\psi_R)_m = (1+k_{s\psi R})\sin\psi_R + b_{s\psi R}$$

$$(40)\quad (\cos\psi_R)_m = (1+k_{c\psi R})\cos\psi_R + b_{c\psi R}$$

3.4 POLYNOMIAL EXPANSION OF COEFFICIENTS

Three hundred forty eight aerodynamic coefficients appear in the equations of motion in Section 3.3. These coefficients are listed in Table 3.2 by giving the index of the coefficient in the row of the appropriate primary symbol and column of the appropriate subscript. For example, coefficient 46 is C_{L_o} and coefficient 112 is $T_{\dot{\beta}_o}$. Coefficients which are usually not significant have their entries enclosed in "( )", i.e., coefficient (3) is C_{x_v}.

Each of these coefficients is evaluated within the program as a polynomial (although in most applications the polynomial will consist of a single, constant term). The procedure by which this is accomplished will appear complex at first; but once it is mastered, the user may appreciate the great flexibility which it gives him.

Table 3.2

Indices of the Aerodynamic Coefficients

PRIMARY SYMBOL	SUBSCRIPT																													SPARES
	o	u	v	w	p	q	r	$\dot\beta_o$	$\dot\beta_{1c}$	$\dot\beta_{1s}$	β_o	β_{1c}	β_{1s}	$\dot\zeta_o$	$\dot\zeta_{1c}$	$\dot\zeta_{1s}$	Ω_R	ζ_o	ζ_{1c}	ζ_{1s}	ψ_R	θ_o	θ_{1c}	θ_{1s}	δ_{TR}	δ_e	δ_a	δ_r	δ_f	
C_x	1	2	(3)	4	(5)	6	(7)																		(8)	(9)	(10)	(11)	(12)	13-15
C_y	16	(17)	18	(19)	20	(21)	22																		23	(24)	(25)	(26)	(27)	28-30
C_z	31	32	(33)	34	(35)	36	(37)																		(38)	(39)	(40)	(41)	(42)	43-45
C_L	46	(47)	48	(49)	50	(51)	52																		53	(54)	(55)	(56)	(57)	58-60
C_M	61	62	(63)	64	(65)	66	(67)																		(68)	(69)	(70)	(71)	(72)	73-75
C_N	76	(77)	78	(79)	80	(81)	82																		83	(84)	(85)	(86)	(87)	88-90
H	91							92	93	94	95	96	97	98	99	100	101	102	103	(104)	(105)	106	107	108						109-110
T	111							112	113	114	115	116	117	118	119	120	121	122	123	(124)	(125)	126	127	128						129-130
Y	131							132	133	134	135	136	137	138	139	140	141	142	143	(144)	(145)	146	147	148						149-150
L	151							152	153	154	155	156	157	158	159	160	161	162	163	(164)	(165)	166	167	168						169-170
M	171							172	173	174	175	176	177	178	179	180	181	182	183	(184)	(185)	186	187	188						189-190
Q	191							192	193	194	195	196	197	198	199	200	201	202	203	(204)	(205)	206	207	208						209-210
H^o	211	212	213	214	215	216	217	281	219	220	221	222	223	224	225	226	227	228	229	(230)	(231)	232	233	234						235
H^{1c}	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	(255)	(256)	257	258	259						260
H^{1s}	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	(280)	(281)	282	283	284						285
Q^o	286	287	288	289	290	291	292	293	244	295	296	297	298	299	300	301	302	303	304	(305)	(306)	307	308	309						310
Q^{1c}	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	(330)	(331)	332	333	334						335
Q^{1s}	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	(355)	(356)	357	358	359						360
Q^2	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	(380)	(381)	382	383	384						385-400

Because of its complexity, the procedure is best introduced by example, Suppose the rolling moment coefficient is to be modeled as

$$C_L = C_{L_o} + C_{L_\beta}(\beta - \beta_o) + C_{L_{\beta^2}}(\beta - \beta_o)^2 + C_{L_{\alpha\delta_{TR}}} \alpha\delta_{TR}$$

where β_o is a constant (not necessarily β at $t = 0$). Let

$$p_1 = C_{L_o}, \quad p_2 = C_{L_\beta}, \quad p_3 = C_{L_{\beta^2}}, \quad p_4 = C_{L_{\alpha\delta_{TR}}}$$

$$z_1 = \alpha, \quad z_{1_o} = 0, \quad z_2 = \beta, \quad z_{2_o} = \beta_o, \quad z_3 = \delta_{TR},$$

$$z_{3_o} = 0$$

Then, Eq. (3.69) is identical to

$$C_L = p_1 + p_2(z_2 - z_{2_o}) + p_3(z_2 - z_{2_o})^2 + p_4 z_1 z_3$$

$$\begin{aligned} C_L = {} & p_1(z_1 - z_{1_o})^o (z_2 - z_{2_o})^o (z_3 - z_{3_o})^o \\ & + p_2(z_1 - z_{1_o})^o (z_2 - z_{2_o})^1 (z_3 - z_{3_o})^o \\ & + p_3(z_1 - z_{1_o})^o (z_2 - z_{2_o})^2 (z_3 - z_{3_o})^o \\ & + p_4(z_1 - z_{1_o})^1 (z_2 - z_{2_o})^o (z_3 - z_{3_o})^1 \end{aligned}$$

$$C_L = \sum_{j=1}^{4} p_j \left\{ \prod_{i=1}^{3} (z_i - z_{i_o})^{n_{ij}} \right\}$$

where the exponents n_{ij} have appropriate values.

The form of Eq. (3.72) which emerged in the above example is similar to the general form used to evaluate each of the thirty-three coefficients. Specifically, the k-th coefficient is computed as

$$C_k = \sum_{j=\ell_1}^{\ell_2} p_j \left\{ \prod_{i=1}^{m} (z_i - z_{i_o})^{n_{ij}} \right\}$$

where

p_j is a parameter which may be identified by the program,

z_i is an "expansion variable,"

z_{i_o} is the reference value of the i-th expansion variable, and

n_{ij} are integer exponents of the expansion variables.

The exponents and the expansion variable reference values are inputs to the program and remain constant throughout the entire run. The parameters are constant throughout an iteration of the optimization search, but they may vary from one iteration to the next if they are identified. The initial values of the parameters are inputs. The expansion variables are functions of the state and control variables and are evaluated repeatedly by the program.

Table 3.3

Expansion Variables

INDEX	SYMBOL	DESCRIPTION	UNITS
1	α	angle-of-attack	rad
2	β	angle of sideslip	rad
3	p	roll rate	rad/sec
4	q	pitch rate	rad/sec
5	r	yaw rate	rad/sec
6	θ_o	collective pitch	rad*
7	θ_{1c}	lateral cyclic pitch	rad*
8	θ_{1s}	longitudinal cyclic pitch	rad*
9	δ_{TR}	tail rotor collective pitch	rad*
10	δ_e	elevator angle	rad*
11	δ_a	aileron angle	rad*
12	δ_r	rudder angle	rad*
13	δ_f	flaperon angle	rad*
14	μ	advance ratio	N.D.
15	u'	longitudinal velocity (normalized)+	N.D.
16	v'	lateral velocity (normalized)	N.D.
17	w'	vertical velocity (normalized)	N.D.

* Recommended units; actual units are determined by the input time histories.

\+ $u' = u/\Omega R$

$v' = v/\Omega R$

$w' = w/\Omega R$

As many as five expansion variables may be selected for a run. Storage and run time constraints make more than five impractical. All the polynomials describing the coefficients must be expressed in terms of the same five (or fewer) expansion variables. The selected variables are specified by input from the list of 17 possible expansion variables in Table 3.3. They may be chosen in any order, but once chosen, the reference values and the exponents must be input such that they correspond to the same order. Of course, the units of the reference values must also be consistent with the units of the corresponding expansion variables.

The equations defining the expansion variables α, β, and μ are:

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right)$$

$$\beta = \tan^{-1}\left(\frac{v}{\sqrt{u^2 + w^2}}\right)$$

$$\mu = \frac{1}{\Omega R}\,(u \cos\theta_R + w \sin\theta_R)$$

3.5 PARAMETERS

NLSCIDNT was written with a capacity for 705 parameters for use in evaluating the equations of motion, the measurment equations, and the expansion variables (see Table 3.4). This is more than sufficient for the present model, so not all parameters are used. Also, some parameters have been reserved for uses which are not currently implemented but are easily added if the need arises. Finally, some parameters may not be identified; for example, the aircraft mass and the air density may not be identified.

Table 3.4

Parameters

INDEX	SYMBOL	DESCRIPTION
1-400		These parameters are available for defining the polynomial expansions of the aerodynamic coefficients
401	$u(0)$	Initial conditions for the state variables
402	$v(0)$	
403	$w(0)$	
404	$p(0)$	
405	$q(0)$	
406	$r(0)$	
407	$\phi(0)$	
408	$\theta(0)$	
409	$\psi(0)$	
410	$\dot{\beta}_o(0)$	
411	$\dot{\beta}_{1c}(0)$	
412	$\dot{\beta}_{1s}(0)$	
413	$\beta_o(0)$	
414	$\beta_{1c}(0)$	
415	$\beta_{1s}(0)$	
416	$\dot{\zeta}_o(0)$	
417	$\dot{\zeta}_{1c}(0)$	
418	$\dot{\zeta}_{1s}(0)$	
419	$\delta\Omega_R(0)$	
420	$\zeta_o(0)$	
421	$\zeta_{1c}(0)$	
422	$\zeta_{1s}(0)$	
423	$\delta\psi_R(0)$	
426	g	Gravitational acceleration (default - 32.174 ft/sec^2)
427	ρ	Air density
428	θ_R	Rotor shaft tilt in X-Z plane, (+) forward

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
429	Ω	Rotor speed
430	R	Rotor radius
431	σ	NC/ R
432	γ	Lock number = $\rho acR^4/I_b$
433	a	2-D lift curve slope
434	m	Mass of vehicle
435	x_{HUB}	Longitudinal distance of rotor hub from vehicle C.G.
436	z_{HUB}	Vertical distance of rotor hub from vehicle C.G.
437	I_x	
438	I_y	
439	I_z	
440	I_{xy}	fuselage moments of inertia (slug-ft^2)
441	I_{xz}	
442	I_{yz}	
443	I_o	blade moment of inertia (slug-ft^2)
444	I_β	blade flap moment of inertia N.D.
445	I_ζ	blade lag moment of inertia N.D.
446	S_β	
447	S_ζ	
448	$I_{\beta\dot{\beta}}$	
449	$I_{\beta\alpha}$	
450	$I_{\beta\dot{\psi}}$	
451	$I_{\dot{\beta}\alpha}$	
452	$I_{\dot{\beta}\psi}$	inertia terms - for definition see page xi, Definitions of Inertial Constants in the final report [10]
453	$I_{\zeta\dot{\zeta}}$	
454	$I_{\zeta\alpha}$	
455	$I_{\zeta\dot{\psi}}$	
456	$I_{\dot{\zeta}\alpha}$	
457	$I_{\dot{\zeta}\psi}$	
458	N_b	Number of blades

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
459	ν_β	Inertia terms
460	ν_ζ	
461	g_s	
		<u>Longitudinal Accelerometer</u>
470	R_x	Noise covariance
471	b_x	Bias
472	k_x	Scale factor error
473	x_{cgx}	X location
474	y_{cgx}	Y location
475	z_{cgx}	Z location
		<u>Lateral Accelerometer</u>
479	R_y	Noise covariance
480	b_y	Bias
481	k_y	Scale factor error
482	x_{cgy}	X location
483	y_{cgy}	Y location
484	z_{cgy}	Z location
		<u>Vertical Accelerometer</u>
488	R_z	Noise covariance
489	b_z	Bias
490	k_z	Scale factor error
491	x_{cgz}	X location
492	y_{cgz}	Y location
493	z_{cgz}	Z location

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
		Vertical Accelerometer (Cont'd)
494		
495		
		Roll Angular Accelerometer
497	$R_{\dot{p}}$	Noise covariance
498	$b_{\dot{p}}$	Bias
499	$k_{\dot{p}}$	Scale factor error
		Pitch Angular Accelerometer
503	$R_{\dot{q}}$	Noise covariance
504	$b_{\dot{q}}$	Bias
505	$k_{\dot{q}}$	Scale factor error
		Yaw Angular Accelerometer
509	$R_{\dot{r}}$	Noise covariance
510	$b_{\dot{r}}$	Bias
511	$k_{\dot{r}}$	Scale factor error
		Roll Rate Gyro
515	R_p	Noise covariance
516	b_p	Bias
517	k_p	Scale factor error

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
		<u>Pitch Rate Gyro</u>
521	R_q	Noise covariance
522	b_q	Bias
523	k_q	Scale factor error
		<u>Yaw Rate Gyro</u>
527	R_r	Noise covariance
528	b_r	Bias
529	k_r	Scale factor error
		<u>Roll Position Gyro</u>
533	R_ϕ	Noise covariance
534	b_ϕ	Bias
535	k_ϕ	Scale factor error
		<u>Pitch Position Gyro</u>
539	R_θ	Noise covariance
540	b_θ	Bias
541	k_θ	Scale factor error

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
		<u>Yaw Position Gyro</u>
545	R_ψ	Noise covariance
546	b_ψ	Bias
547	k_ψ	Scale factor error
548		
549		
		<u>Angle-of-Attack Vane</u>
551	R_α	Noise covariance
552	b_α	Bias
553	k_α	Scale factor error
554	$x_{cg\alpha}$	X location
555	$y_{cg\alpha}$	Y location
556	$z_{cg\alpha}$	Z location
557		
		<u>Sideslip Vane</u>
558	R_β	Noise covariance
559	b_β	Bias
560	k_β	Scale factor error
561	$x_{cg\beta}$	X location
562	$y_{cg\beta}$	Y location
563	$z_{cg\beta}$	Z location
564		
		<u>Pitot Tube</u>
565	R_{VT}	Noise covariance
566	b_{VT}	Bias
567	k_{VT}	Scale factor error
568	x_{VT}	X location
569	y_{VT}	Y location
570	z_{VT}	Z location

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
		Pitot Tube (Cont'd)
571	p_o	Reference air density
572	v_s	Velocity of sound
		Longitudinal Velocity Measurement
574	R_u	Noise covariance
575	b_u	Bias
576	k_u	Scale factor error
		Lateral Velocity Measurement
578	R_v	Noise covariance
579	b_v	Bias
580	k_v	Scale factor error
		Vertical Velocity Measurement
582	R_w	Noise covariance
583	b_w	Bias
584	k_w	Scale factor error
585		
586	$\tilde{x}_{ref}$	Horizontal reference distance
588	$\tilde{z}_{ref}$	Vertical reference distance
		Rotor Longitudinal Force Measurement
590	R_{xR}	Noise covariance
591	b_{xR}	Bias
592	k_{xR}	Scale factor error

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
		Rotor Lateral Force Measurement
594	R_{y_R}	Noise covariance
595	b_{y_R}	Bias
596	k_{y_R}	Scale factor error
		Rotor Vertical Force Measurement
598	R_{z_R}	Noise covariance
599	b_{z_R}	Bias
600	k_{z_R}	Scale factor error
		Rotor Roll Moment Measurement
602	R_{L_R}	Noise covariance
603	b_{L_R}	Bias
604	k_{L_R}	Scale factor error
		Rotor Pitch Moment Measurement
606	R_{M_R}	Noise covariance
607	b_{M_R}	Bias
608	k_{M_R}	Scale factor error
		Rotor Yaw Moment Measurement
610	R_{N_R}	Noise covariance
611	b_{N_R}	Bias
612	k_{N_R}	Scale factor error
614	ϕ_1	Euler angle
615	$\Delta\phi$	Increment
616	θ_1	Euler angle
617	$\Delta\theta$	Increment

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
618	$\beta_{o\ ref}$	
619	$\beta_{1c\ ref}$	Blade flapping
620	$\beta_{1s\ ref}$	
		<u>Blade 1 Flapping Measurement</u>
622	$R_{\beta 1}$	Noise covariance
623	$b_{\beta 1}$	Bias
624	$k_{\beta 1}$	Scale factor error
		<u>Blade 2 Flapping Measurement</u>
626	$R_{\beta 2}$	Noise covariance
627	$b_{\beta 2}$	Bias
628	$k_{\beta 2}$	Scale factor error
		<u>Blade 3 Flapping Measurement</u>
630	$R_{\beta 3}$	Noise covariance
631	$b_{\beta 3}$	Bias
632	$k_{\beta 3}$	Scale factor error
		<u>Blade 4 Flapping Measurement</u>
634	$R_{\beta 4}$	Noise covariance
635	$b_{\beta 4}$	Bias
636	$k_{\beta 4}$	Scale factor error
		<u>Blade 5 Flapping Measurement</u>
638	$R_{\beta 5}$	Noise covariance
639	$b_{\beta 5}$	Bias
640	$k_{\beta 5}$	Scale factor error

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
		<u>Blade 6 Flapping Measurement</u>
642	$R_{\beta 6}$	Noise covariance
643	$b_{\beta 6}$	Bias
644	$k_{\beta 6}$	Scale factor error
645		
		<u>Blade 7 Flapping Measurement</u>
646	$R_{\beta 7}$	Noise covariance
647	$b_{\beta 7}$	Bias
648	$k_{\beta 7}$	Scale factor error
649		
650	ζ_{o} ref	
651	ζ_{1c} ref	Lag references
652	ζ_{1s} ref	
653		
		<u>Blade 1 Lag Measurement</u>
654	$R_{\zeta 1}$	Noise covariance
655	$b_{\zeta 1}$	Bias
656	$k_{\zeta 1}$	Scale factor error
657		
		<u>Blade 2 Lag Measurement</u>
658	$R_{\zeta 2}$	Noise covariance
659	$b_{\zeta 2}$	Bias
660	$k_{\zeta 2}$	Scale factor error
661		
		<u>Blade 3 Lag Measurement</u>
662	$R_{\zeta 3}$	Noise covariance
663	$b_{\zeta 3}$	Bias
664	$k_{\zeta 3}$	Scale factor error
665		

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
		<u>Blade 4 Lag Measurement</u>
666	$R_{\zeta 4}$	Noise covariance
667	$b_{\zeta 4}$	Bias
668	$k_{\zeta 4}$	Scale factor error
669		
		<u>Blade 5 Lag Measurement</u>
670	$R_{\zeta 5}$	Noise covariance
671	$b_{\zeta 5}$	Bias
672	$k_{\zeta 5}$	Scale factor error
673		
		<u>Blade 6 Lag Measurement</u>
674	$R_{\zeta 6}$	Noise covariance
675	$b_{\zeta 6}$	Bias
676	$k_{\zeta 6}$	Scale factor error
677		
		<u>Blade 7 Lag Measurement</u>
678	$R_{\zeta 7}$	Noise covariance
679	$b_{\zeta 7}$	Bias
680	$k_{\zeta 7}$	Scale factor error
681		
682	$R_{S\psi_R}$	Noise covariance
683	$k_{S\psi_R}$	Scale factor error
684	$b_{S\psi_R}$	Bias
685		

Table 3.4 (Continued)

INDEX	SYMBOL	DESCRIPTION
686	$R_{S\psi_R}$	Scale factor error
687	$k_{S\psi_R}$	Bias
688	$b_{S\psi_R}$	Noise covariance
689		
701	z_{1_o}	Reference value* for expansion variable z_1
702	z_{2_o}	Reference value* for expansion variable z_2
703	z_{3_o}	Reference value* for expansion variable z_3
704	z_{4_o}	Reference value* for expansion variable z_4
705	z_{5_o}	Reference value* for expansion variable z_5
		* (bias)

Generally, most of the parameters which are identified in a run are those used in forming the polynomial expansions of the aerodynamic coefficients. Parameters 1 through 400 are reserved for this purpose. However. the user should always use the low end of this range, when all locations are not needed, as this will reduce the run time. Chapter IV will explain in more detail how these expansions are accomplished.

CHAPTER IV
PROGRAM INPUT

Two forms of input are used by the NLSCIDNT program. Information about the parameters and the selection of various options are read from cards. The time histories of the measurements and controls are read from a mass storage device, such as magnetic tape or disk.

4.1 CARD INPUT

All data cards are read by subroutine INPUT. The forms of the input fall into several types, which are listed in Table 4.1. The input sequence is as follows:

(1) One card of type 1. The information on this card is printed at the top of every page of printout.

(2) One card of type 2. For an explanation of "outer" and "inner" iterations, see Section 5.2.2. A "prediction run" is one in which the trajectory is simulated using the control input time histories and input parameter values; no identification is attempted.

(3) One card of type 3. This card sets various option flags. If IPLØT = 2 or 3, the user must provide a tape on logical unit 3 (see Section 5.3).

(4) Two or more cards of type 4. These cards tell the program which states are to be integrated from the equations of motion, which states are to be looked up from an input time history, which measurements are used and their order, and which expansion variables are used and their order. If no states are to be looked-up, then a card beginning with "READ" is not necessary. Any state variable whose value does not appear on a "STATES" or "READ" card will automatically be held constant by the program at its initial value, which is one of the input parameters. These cards may appear in any order except the "*" card must be last.

Table 4.1

Input Card Types

CARD TYPE	COLUMNS	FORMAT	VARIABLE NAME	DESCRIPTION
1	1-60	15A4	TITLE	Program identification information.
2	4-5	I2	K1MAX	Maximum number of outer iterations > 0, normally = 0, prediction run =-1, only 0th iteration will be computed (usually selected for diagnostic purposes only)
	9-10	I2	K2MAX	Maximum number of inner iterations > 0, normally = 0, prediction run
	12-15	I4	NN	Number of data sample points ($\leq$ 501)
	16-25	E10.3	DELTA	Time interval between sample points of the time history data (in sec.)
	26-35	E10.3	RELER1	Relative error bound for $\hat{x}$ and $\partial\hat{x}/\partial\hat{\theta}$ allowed in integration algorithm (defaults to 10^{-5}).
	36-45	E10.3	ABSER1	Absolute error bound (defaults to 1.7×10^{-4}).
	45-65	E10.3	FMARQ	Initial value of Marquardt parameter (Defaults to 1.)
	56-65	E10.3	XMARQ	Factor by which the Marquardt parameter will be increased or decreased as search proceeds. (Defaults to 2.)
	66-70	I5	LLL	=1, the Marquardt parameter is computed as the current value of FMARQ times the largest element of the gradient ($\partial J/\partial\theta$). =0, the Marquardt parameter is FMARQ.

Table 4.1 (Continued)

CARD TYPE	COLUMNS	FORMAT	VARIABLE NAME	DESCRIPTION
3	5	I1	IPLØT	=3, if both printer plots and a tape storing the time history data are to be made =2, if only the tape is to be made =1, if only the printer plots are to be made =0, if neither
	10	I1	IPLTC	=1, if printer plots of control input time histories are to be made =0, if not (IPLTC is ignored if IPLØT = 0 or 2)
	12-15	I4	IDATA	=+k, then the first k sample points of the measurement and control time histories read into the program will be printed =-k, then every kth sample point will be printed = 0, if they are not to be printed
	20	I1	IPRNT	Sets level of detail of the diagnostic printout (see Section 4.2) =3, highest level =2, medium level =1, lowest level =0, normal printout only
	22-25	I4	INCPR2	=k, if $\hat{x}$, $\partial\hat{x}/\partial\hat{\theta}$, $\hat{y}$, $\partial\hat{y}/\partial\hat{\theta}$, and y are to be printed every kth sample point
	26-30		INCPR3	Default value is 50. INCPR3 is ignored if IPRNT < 3.
	33-35	I3	INCPLT	=k, every kth data point is plotted in the printer plots (default =1)

Table 4.1 (Continued)

CARD TYPE	COLUMNS	FORMAT	VARIABLE NAME	DESCRIPTION
3	40	I1	IINFØ	=1, a priori information matrix is to be read =0, a priori information matrix is not to be read but assumed zero
	45	I1	IRCMP	=0, estimated noise covariance, $\hat{R}$, to be computed as the diagonal of the innovations covariance (see Appendix A) =1, $\hat{R}$ computed from parameters
	50	I1	IRVARY	=1, $\hat{R}$ is assumed nonstationary (it is a function of time which user must code into subroutine MEAS). =1, R is assumed stationary (constant throughout an iteration).
	52-55	I4	MAXNM1	Maximum number of integration steps permitted in one sample interval.
	56-60	I5	MCYCLE	If non-zero, it is the number of parameters to be identified on this run.
	61-65	I5	ILVL1	Flag to use linearized state and measurement equations if ≠ 0.
4	1-10	A8,2X	LTYPE	= STATES, if the card lists the input names of states which are to be integrated = READ, if the card lists the input names of the states which will be looked-up from input time histories = MEASURE, if the card lists the input names of the measurement variables = EXP VARS, if the card lists the input names of the expansion variables = *b, if no more cards of this type are to be read ("b" indicates a blank). Note that only the first two characters are actually read; these words must be left justified.

Table 4.1 (Continued)

CARD TYPE	COLUMNS	FORMAT	VARIABLE NAME	DESCRIPTION
4	11-80	14(A3,2X)	(LL(J), J=1,14)	Input names of state, measurement, or expansion variables as listed in Table 4.2. Note that only the first three characters are actually read; these words must be left justified. Blank words may appear between input names.
5	1	A1	ECHK	= blank, if this card contains parameter information = *, if no more cards of this type are to be read
	2-4	I3	J1	Parameter index (see Table 3.4)
	5	A1	J2	= *, if this parameter is to be identified = blank, if not
	6-11	A6	PLABJ1	Label for the parameter for printout purposes
	12-29	D18.0	PJ1	Initial value of the parameter
	31-32	I3	II(1)	=k, this parameter appears in the polynomial expansion of the kth aero coefficient (see Table 3.2), Ignored if J1 > 400.
	34-38	5I1	II(J), J=2,...,6	Exponents of the expansion variables in the aero coefficient polynomial term containing this parameter.
	51-60	D10.0	PLJ1	Lower bound of the parameter.
	61-70	D10.0	PUJ1	Upper bound of the parameter. (If both PLJ1 = 0 and PUJ1 = 0, then PLJ1 defaults to -10^{27} and PUJ1 defaults to $+10^{27}$).

Table 4.1 (Continued)

CARD TYPE	COLUMNS	FORMAT	VARIABLE NAME	DESCRIPTION
6	1-80	8F10.0	INFØ(J,K) K=1,...,m	Jth row of the a priori information matrix; continue on another card if necessary (m = number of parameters being identified).
7	1	A1	JA	= T, the label which follows on this card is for time (the independent variable) = Y, the label is for a measurement variable = U, the label is for a control variable = X, the label is for a state variable (only look-up states are affected)
	3-4	I2	J	The index number of the variable. If JA = T, then J is ignored.
	7-30	6A4	NAME	A label of six 4-character words. Usually the variable name appears in the first 3 words and its units in the last 3 words.

(5) As many cards of type 5 as needed to input all information about the parameters; each card describes one parameter. If a parameter is zero and is not identified, no card need be inserted for it. A card with "*" in column 1 tells the program that all cards of this type have been read. The system of units for the parameter values is feet, pounds, slugs, seconds, and radians or meters, newtons, kilograms, seconds, and radians. It is the responsibility of the user to maintain consistent units throughout.

(6) Cards of type 6 are read only if IINFØ = 1 on card type 3. These cards are used to read the a priori information matrix. This matrix must be m x m, where m is the number of parameters being identified. Furthermore, the order of the rows and columns must correspond to the order of the identified parameters as listed in the cards of type 5. The matrix is read by rows; each row begins on a new card and continues on more cards as needed.

(7) As many cards of type 7 as needed to alter the labels of time, measurements, and controls in the printout and printer plots of the time histories. These labels default to the output labels listed in Table 4.2. Therefore, the most common use of these cards is to change the units shown in the labels. If none of the labels need altering, then no cards of type 7 are necessary.

4.2 EXAMPLE INPUT DECK

As an illustration of the card input, consider the following example. The data processed were simulated from a rotorcraft in level flight at 100 knots at sea level. Longitudinal cyclic pitch control inputs excited the longitudinal dynamics. However, some perturbations in the lateral dynamics were observed and of enough significance that they could not be neglected. A NLSCIDNT run was made to identify three aerodynamic derivatives.

Figure 4.1 shows the input card deck for this example run. The first card contains the title of the run. The second card specifies one outer iteration, a maximum of four inner iterations permitted, that there are 101 sample points, and that 0.02 second

Table 4.2
Input Names and Output Labels of Program Variables

	SYMBOL	INPUT NAME[1]	OUTPUT LABEL[2]
TIME	t	---	TIME (SECONDS)
STATES	u	U	LONGIT VEL (M/SEC)
	v	V	LATERAL VEL (M/SEC)
	w	W	VERTICAL VEL (M/SEC)
	p	P	ROLL RATE (RAD/SEC)
	q	Q	PITCH RATE (RAD/SEC)
	r	R	YAW RATE (RAD/SEC)
	ϕ	PHI	ROLL ANGLE (RAD)
	θ	THET	PITCH ANGLE (RAD)
	ψ	PSI	YAW ANGLE (RAD)
	$\dot{\beta}_o$	BOD	BETA DØT (RAD/SEC)
	$\dot{\beta}_{1c}$	BCD	BETA 1c DØT (RAD/SEC)
	$\dot{\beta}_{1s}$	BSD	BETA 1s DØT (RAD/SEC)
	β_o	BO	BETA 0 (RAD)
	β_{1c}	BC	BETA 1c (RAD)
	β_{1s}	BS	BETA 1s (RAD)
	$\dot{\zeta}_o$	ZOD	ZETA 0 DØT (RAD)
	$\dot{\zeta}_{1c}$	ZCD	ZETA 1c DØT (RAD/SEC)
	$\dot{\zeta}_{1s}$	ZSD	ZETA 1s DØT (RAD/SEC)
	Ω_s	OME	OMEGA PERT. (RAD/SEC)
	ζ_o	ZO	ZETA 0 (RAD)
	ζ_{1c}	ZC	ZETA 1c (RAD)
	ζ_{1s}	ZS	ZETA 1s (RAD)
	ψ_s	RSP	PSI-S PERT. (RAD)

Table 4.2 (Continued)

	SYMBOL	INPUT NAME[1]	OUTPUT LABEL[2]	
MEASUREMENTS	a_X	AX	AXIAL ACC	(M/SEC**2)
	a_Y	AY	LAT ACC	(M/SEC**2)
	a_X	AZ	VERT ACC	(M/SEC**2)
	$\dot{p}_m$	PD	ROLL ACC	(RAD/SEC**2)
	$\dot{q}_m$	QD	PITCH ACC	(RAD/SEC**2)
	$\dot{r}_m$	RD	YAW ACC	(RAD/SEC**2)
	p_m	P	ROLL RATE	(RAD/SEC**2)
	q_m	Q	PITCH RATE	(RAD/SEC**2)
	r_m	R	YAW RATE	(RAD/SEC**2)
	ϕ_m	PHI	ROLL ANG	(RADIANS)
	θ_m	THE	PITCH ANG	(RADIANS)
	ψ_m	PSI	YAW ANG	(RADIANS)
	α_m	ALP	ALPHA	(RADIANS)
	β_m	BET	BETA	(RADIANS)
	V_m	VEL	TOTAL VELOCITY	(M/SEC)
	u_m	U	LONG VEL	(M/SEC)
	v_m	V	LAT VEL	(M/SEC)
	w_m	W	VERT VEL	(M/SEC)
	X_{R_m}	XR	ROTOR LONG FORCE	(N)
	Y_{R_m}	YR	ROTOR LAT FORCE	(N)
	Z_{R_m}	ZR	ROTOR VERT FORCE	(N)
	L_{R_m}	LR	ROTOR ROLL MOMENT	(N-M)
	M_{R_m}	MR	ROTOR PITCH MOMENT	(N-M)
	N_{R_m}	NR	ROTOR YAW MOMENT	(N-M)
	β_{1_m}	B1	FLAP ANGLE-BLADE 1	(RAD)
	β_{2_m}	B2	FLAP ANGLE-BLADE 2	(RAD)
	β_{3_m}	B3	FLAP ANGLE-BLADE 3	(RAD)

Table 4.2 (Continued)

	SYMBOL	INPUT NAME[1]	OUTPUT LABEL[2]
M E A S U R E M E N T S	β_{4_m}	B4	FLAP ANGLE-BLADE 4 (RAD)
	β_{5_m}	B5	FLAP ANGLE-BLADE 5 (RAD)
	β_{6_m}	B6	FLAP ANGLE-BLADE 6 (RAD)
	β_{7_m}	B7	FLAP ANGLE-BLADE 7 (RAD)
	ζ_{1_m}	Z1	LAG ANGLE-BLADE 1 (RAD)
	ζ_{2_m}	Z2	LAG ANGLE-BLADE 2 (RAD)
	ζ_{3_m}	Z3	LAG ANGLE-BLADE 3 (RAD)
	ζ_{4_m}	Z4	LAG ANGLE-BLADE 4 (RAD)
	ζ_{5_m}	Z5	LAG ANGLE-BLADE 5 (RAD)
	ζ_{6_m}	Z6	LAG ANGLE-BLADE 6 (RAD)
	ζ_{7_m}	Z7	LAG ANGLE-BLADE 7 (RAD)
	$(\cos \psi_R)_m$	COS	COS ROTOR AZIMUTH (RAD)
	$(\sin \psi_R)_m$	SIN	SIN ROTOR AZIMUTH (RAD)

Table 4.2 (Continued)

	SYMBOL	INPUT NAME[1]	OUTPUT LABEL[2]
CONTROLS	θ_o	TO	COLL PITCH (RADIANS)
	θ_{1c}	TC	LAT CY PITCH(RADIANS)
	θ_{1s}	TS	LON CY PITCH(RADIANS)
	δ_{TR}	DTR	TAIL COLL (RADIANS)
	δ_e	DE	ELEVATOR (RADIANS)
	δ_a	DA	AILERON (RADIANS)
	δ_r	DR	RUDDER (RADIANS)
	δ_f	DF	FLAPERON (RADIANS)
EXPANSION VARIABLES	α	ALPH	ALPHA
	β	BETA	BETA
	p	P	ROLL RATE
	q	Q	PITCH RATE
	r	R	YAW RATE
	θ_o	TO	COLLECTIVE
	θ_{1c}	TC	LAT CYCLIC
	θ_{1s}	TS	LONG CYCLIC
	δ_{TR}	DTR	TAIL ROTOR
	δ_e	DE	ELEVATOR
	δ_a	DA	AILERON
	δ_r	DR	RUDDER
	δ_f	DF	FLAPERON

Table 4.2 (Concluded)

	SYMBOL	INPUT NAME[1]	OUTPUT LABEL[2]
EXPANSION VARIABLES	μ	MU	MU
	u	U	u (NORM)
	v	V	v (NORM)
	w	W	w (NORM)

[1]The underlined characters of the input name are what the program actually reads.

[2]"MS" in the measurement labels indicates the measured quantity; "ES" appears instead in the label to indicate the estimated quantity.

CARD TYPE	CARD NO.	DATA CARDS	COMMENTS
1	1.	LATERAL CRUISE DYNAMICS EXAMPLE	title
2	2.	1 4 101 0.02	max iterations, max step cuts, no. of sample points, and sample interval
3	3.	1 1 11 3 25 25 0 0 1 0	option flags
4	4.	STATES: V P R PHI	
	5.	READ: W Q THETA	
	6.	MEASURE: AY BETA P R PHI	specification of variables used in this run
	7.	EXP VARS: BETA DSP	
	8.	*	
5	9.	1 CY 0. 6	
	10.	2 CYV -.004268 7	
	11.	3 CYP -.08892 8	
	12.	4 CYR 0.4447 9	
	13.	5 CYDA .003121 10	parameter number
	14.	6 CYDR 0.3468 11	
	15.	7 CLO 0. 17	parameter name
	16.	8 CLV -.0005354 18	parameter value
	17.	9*CLP -0.55 19 -10000000. 0.	lower bound; upper bound
	18.	10 CLR 0.1542 20	
	19.	11 CLDA 0.1852 21	
	20.	12 CLDR .02703 22	
	21.	13 CNO 0. 28	
	22.	14 CNV .000553 29	index of coefficient in whose expansion this parameter appears
	23.	15 CNP -.06153 30	
	24.	16 CNR -0.1493 31	
	25.	17 CNDA .005994 32	
	26.	18 CNDR -0.1321 33	
	27.	19 CNB3 -2.566 26 30	exponents of expansion variables
	28.	20*CLDS 0. 17 01	
	29.	21*CLDS3 0. 17 03	
	30.	81 RHO .0023769	indicates the parameter is to be identified
	31.	82 CBAR 6.5	
	32.	83 S 420.	
	33.	84 B 65.	
	34.	85 MASS 342.	
	35.	86 IX 16800.	
	36.	87 IY 24900.	
	37.	88 IZ 37550.	
	38.	89 IXZ 0.	
	39.	95 VSND 1117.	
	40.	101 UO 179.75	
	41.	103 WO 9.42	
	42.	*	
6	43.	7762.4 2019.1 52.651	
	44.	2019.1 69264. 176.60	a priori information matrix
	45.	52.651 176.6 0.47228	
7	46.	T TIME (SEC) REF TO T=6.3	specification of output labels
	47.	X(7) PHI (RAD)	

Figure 4.1 Example Input Card Deck

is the sample interval. The third card asks for printer plots, for the first 11 sample points of the input time histories to be printed, for the highest (third) level of diagnostic printout at intervals of 25 sample points, and for the second level of diagnostic printout at intervals of 25 sample points. It also specifies that an a priori information matrix is not to be read, that the noise covariance $\hat{R}$ be computed from the parameters, and that $\hat{R}$ is constant with respect to time. The maximum allowable number of integration steps per sample interval is set to default to 50.

The fourth card tells the program which states are to be integrated. The states which do not appear in the fourth card will be held constant at their initial values. The fifth card tells which states are to be read in from the test data.

The sixth card specifies the measurements available to the run and their order. The seventh card indicates the expansion variables.

The first 21 parameters (specified on cards 9 through 42) are used in the expansions of the aero coefficients which appear in the equations of motion.

The parameters which are to be identified are P(9), P(20), and P(21). P(16) is bounded such that the identified value will not be permitted to become positive.

Cards 43 to 45 contain the a priori information matrix. Cards 46 to 47 give new labels to time and Φ.

4.3 MASS STORAGE INPUT

The time histories of the measurements, controls, and states which are looked-up rather than integrated or fixed are read by subroutine INREAD. The subroutine INREAD supplied with the program assumes these data are stored on a mass storage device (tape, disk, etc.) referenced by FØRTRAN logical unit 2 and read by an unformatted READ statement:

```
      DØ 10 K=1,NN
   10 READ(2,END=900) T(K),(Y(J,K),J=1,NP),(U(J,K),J=1,NQR)
```

where

NN	=	number of sample points,
T(K)	=	the time at the K-th sample point,
Y(J,K)	=	value of the J-th measurement at the K-th sample point,
NP	=	number of measurement variables used in the run,
U(J,K)	=	value of the J-th control variable or a state variable at the K-th sample time,
NQR	=	number of control variables (4) plus the number of look-up states (those not integrated or fixed)

T, U, and Y are single precision arrays.

The user must ensure that the time history data conform to these rules:

(1) The order of the measurements must be the same as their specified order in the card input (card type 4 in Table 4.1).

(2) The controls must always be in the order of:

- collective pitch
- lateral cyclic pitch
- longitudinal cyclic pitch
- tail rotor pitch
- elevator
- aileron
- rudder
- spoiler

(as they appear in Table 3.1); each must be included, even if it is always zero.

(3) Look-up state variables must be appended as if they were additional controls. Their order must be the same as their specified order in the card input (card type 4 in Table 4.1).

(4) No more than <u>four</u> look-up states are permitted.

For the example run described in Section 4.2, the k-th logical record of the time history data must be:

$$t_k,\ a_Y(t_k),\ \beta_m(t_k),\ p_m(t_k),\ r_m(t_k),\ \phi_m(t_k),\ \delta_e(t_k),$$

$$\delta_a(t_k),\ \delta_r(t_k),\ \delta_{sp}(t_k),\ w(t_k),\ q(t_k),\ \theta(t_k)$$

The user will almost certainly have to rearrange his data to conform to the structure required by this subroutine. If he believes he will process the same data repeatedly, it would be worthwhile to perform this rearrangement once before the first NLSCIDNT run and take advantage of the efficiency of this INREAD. However, if the data will be processed only a very few times, he may choose instead to write his own subroutine INREAD, which will accept the data in their original form. Figure 4.2 shows the source code of the INREAD subroutine supplied with the program, which may serve as a model for the user's own.

```
      SUBROUTINE INREAD (U,NO,Y,NP,T,NN,NOR)
      DIMENSION U(NO,NN),Y(NP,NN),T(NN)
      COMMON /COMIOU/ LO,LI,INTAPE
C
      DO 10 K=1,NN
      READ(INTAPE,ERR=900,END=902) T(K),(Y(J,K),J=1,NP),(U(J,K),J=1,NOR)
   10 CONTINUE
C
   99 RETURN
C
  900 WRITE(LO,901) K,T(K-1)
  901 FORMAT(1H0,'*** ERROR IN SUBROUTINE INREAD.  DEVICE ERROR DETECTED
     1 WHILE ATTEMPTING TO READ DATA POINT ',I4,'.'/5X,'THE TIME AT THE
     2PREVIOUS POINT, T(K-1), WAS ',1PE14.6)
      STOP
  902 WRITE(LO,903) K,T(K-1)
  903 FORMAT(1H0,'*** ERROR IN SUBROUTINE INREAD.  END OF FILE ENCOUNTER
     1ED WHILE ATTEMPTING TO READ DATA POINT ',I4,'.'/5X,'THE TIME AT TH
     2E PREVIOUS POINT, T(K-1), WAS ',1PE14.6)
      STOP
C
      END
```

Figure 4.2 Subroutine INREAD Source Code

V. PROGRAM OUTPUT

Most of the program's output is written by the printer, including optional plots. The program also has an option for writing data on a mass storage device, which gives the user the information necessary to make off-line, pen-and-ink plots (e.g. Calcomp plots).

5.1 PRINTED OUTPUT

The information printed by the program falls into six classifications. Not all classifications may appear since the user has considerable control over the detail of the printout through his selection of various options on input card type 3. The six classes are:

(1) repetition of the inputs to the program,

(2) information on the integration of the differential equations,

(3) information on the Levenberg-Marquardt iterative search,

(4) information on the steps taken in the identified parameters and their error standard deviations,

(5) the final parameter estimates (always given), and

(6) printer plots showing: (a) the fit of the estimated measurements to the actual measurements (IPLOT = 1 or 3), and (b) the control time histories (IPLTC = 1).

5.1.1 Example of Printed Output

Figure 5.1 consists of selected pages from the printout of the same example run whose input was discussed in Section 4.2. It contains examples of all the available printed output.

The information in Figure 5.1a is a repetition of the input and is self-explanatory except for the flag arrays. ISFLAG(J) contains: 0 if the jth state variable will be fixed to its initial

```
*************************************************************
*                                                           *
*                  NONLINEAR SCIDNT PROGRAM                 *
*                  ------------------------                 *
*                                                           *
*        MAXIMUM-LIKELIHOOD PARAMETER IDENTIFICATION        *
*     OF NONLINEAR, 6-DEGREE-OF-FREEDOM AIRCRAFT MODEL      *
*                                                           *
*  WRITTEN FOR:                    WRITTEN BY:              *
*  NAVAL AIR TEST CENTER           SYSTEMS CONTROL, INC.    *
*  PATUXENT RIVER, MD.             PALO ALTO, CAL.          *
*                                                           *
*************************************************************

THE PROGRAM HAS SET TO THE FOLLOWING DIMENSIONS...
                                               WAS LIMIT EXCEEDED?
    3  PARAMETERS TO BE IDENTIFIED . . .          NO
    8  STATES . . . . . . . . . . . . . .         NO
    5  MEASUREMENTS . . . . . . . . . . .         NO
    7  CONTROLS . . . . . . . . . . . . .         NO
  205  PARAMETERS IN MODEL . . . . . . . .        NO

THE SAMPLE TIME INTERVAL IS      .02000
THE MAXIMUM NUMBER OF STEPS IN ONE SAMPLE INTERVAL IS  50
THE RELATIVE ERROR TOLERANCE IS   1.00000D-005
THE ABSOLUTE ERROR TOLERANCE IS   1.70000D-004

MAXIMUM NUMBER OF ITERATIONS IS    1
MAXIMUM NUMBER OF STEP CUTS IS     4

EXPANSION VARIABLES USED IN THIS RUN...
  Z(1) = BETA
  Z(2) = SPOILER

THE FLAG ARRAYS FOR THIS RUN ARE...
  ISFLAG =   0  1 -1  2 -2  3  4 -3  0
  NSFLAG =   2  4  6  7
  NMFLAG =   2 14  7  9 10
  NZFLAG =   2  9
```


Figure 5.1a Example NLSCIDNT Printout

value, +k to indicate it is the kth element in the vector of state variables to be integrated, and -k to indicate it is the kth element in the vector of look-up states. NSFLAG contains the indices of the states to be integrated. NMFLAG contains the indices of the measurement variables used in the run. NZFLAG contains the indices of the expansion variables used in the run. NZFLAG(1) will default to 1 if no expansion variables are used.

Figure 5.1(b) shows the parameter input information. If a parameter does not appear in the list, its value is zero and it is not identified. Figure 5.1(c) presents a sample of the input time history data. Only the first 11 sample points were printed because IDATA was set to 11 by the input deck (refer to card type 3 in Table 4.1 and Figure 4.1).

The CV matrix in Figure 5.1d is

$$CV = \sum_{i=1}^{N} \begin{bmatrix} y_i \\ u_i \\ x_{L_i} \end{bmatrix} [y_i^T \; u_i^T \; x_{L_i}^T] \qquad (5.1)$$

where

y_i = measurement vector at $t = t_i$,

u_i = control vector at $t = t_i$,

x_{L_i} = vector of look-up states at $t = t_i$,

N = total number of sample points.

Eleven elements are zero because all controls except longitudinal cyclic pitch are zero for all N points and because the four look-up states are not used.

The IU array contains the indices of non-zero control variables. Look-up states would be treated as additional controls.

*** INITIAL PARAMETER VALUES ***

NO	ID	NAME	COFF	EXPONENTS	PARAM VALUES	LOWER BOUND	UPPER BOUND
1		CY	6		0.00000	0.00000	0.00000
2		CYV	7		-4.26800-003	0.00000	0.00000
3		CYP	8		-8.49200-002	0.00000	0.00000
4		CYR	9		4.44700-001	0.00000	0.00000
5		CYDA	10		3.12100-003	0.00000	0.00000
6		CYDR	11		3.46800-001	0.00000	0.00000
7		CL0	17		0.00000	0.00000	0.00000
8		CLV	18		-5.35400-004	0.00000	0.00000
9	*	CLP	19		-5.50000-001	-1.00000+007	0.00000
10		CLR	20		1.54200-001	0.00000	0.00000
11		CLDA	21		1.85200-001	0.00000	0.00000
12		CLDR	22		2.70300-002	0.00000	0.00000
13		CN0	28		0.00000	0.00000	0.00000
14		CNV	29		5.53000-004	0.00000	0.00000
15		CNP	30		-6.15300-002	0.00000	0.00000
16		CNR	31		-1.49300-001	0.00000	0.00000
17		CNDA	32		5.09400-003	0.00000	0.00000
18		CNDR	33		-1.32100-001	0.00000	0.00000
19		CNB3	28	30000	-2.58600+000	0.00000	0.00000
20	*	CLDS	17	01000	0.00000	-1.00000+027	1.00000+027
21	*	CLDS3	17	03000	0.00000	-1.00000+027	1.00000+027
81		RHO			2.37690-003	0.00000	0.00000
82		CBAR			6.50000+000	0.00000	0.00000
83		S			4.20000+002	0.00000	0.00000
84		B			6.50000+001	0.00000	0.00000
85		MASS			3.42000+002	0.00000	0.00000
86		IX			1.68000+004	0.00000	0.00000
87		IY			2.49000+004	0.00000	0.00000
88		IZ			3.75500+004	0.00000	0.00000
89		IXZ			0.00000	0.00000	0.00000
95		VSND			1.11700+003	0.00000	0.00000
101		U0			1.79750+002	0.00000	0.00000
103		W0			9.42000+000	0.00000	0.00000

INFO

	1	2	3
1	7.7624+003	2.0191+003	5.2651+001
2	2.0191+003	6.9264+004	1.7660+002
3	5.2651+001	1.7660+002	4.7228-001

Figure 5.1b

LATERAL CRUISE DYNAMICS EXAMPLE

NLSCIDNT PROGRAM

TIME (SE	LAT ACCEL VERTICAL V	BETA PITCH RATE	ROLL RATE PITCH ANGL	YAW RATE	ROLL ANG	ELEVATOR	AILERON	RUDDER	SPOILER
.000	1.132608-01 9.420174+00	-4.361622-03 0.000000	-3.533576-02 0.000000	-8.882403-03	2.360781-03	0.000000	0.000000	0.000000	0.000000
.020	7.832291-02 9.420193+00	-4.357742-03 2.177622-07	8.338030-03 2.251232-09	7.381899-03	-2.955314-03	0.000000	0.000000	1.253332-02	0.000000
.040	5.009891-01 9.420212+00	4.527394-03 4.127460-07	-2.009307-02 8.749587-09	2.015344-02	1.741394-02	0.000000	0.000000	2.486899-02	0.000000
.060	2.066717-01 9.420230+00	3.200351-03 5.450003-07	-6.304149-03 1.972265-08	1.074564-02	3.947345-03	0.000000	0.000000	3.681245-02	0.000000
.080	6.687426-01 9.420247+00	3.093622-03 5.267060-07	-1.530348-02 3.074583-08	-1.566161-02	1.564400-02	0.000000	0.000000	4.817537-02	0.000000
.100	1.020005+00 9.420261+00	4.453213-03 2.196172-07	2.764230-03 6.357972-08	-3.103635-02	4.167230-04	0.000000	0.000000	5.877852-02	0.000000
.120	1.069096+00 9.420266+00	8.095325-03 -5.550647-07	1.307635-02 1.088202-07	-4.549416-02	-5.425269-03	0.000000	0.000000	6.845471-02	0.000000
.140	1.070581+00 9.420259+00	3.889000-03 -1.996287-06	1.190308-02 1.830212-07	1.218867-03	-8.201941-03	0.000000	0.000000	7.705132-02	0.000000
.160	9.898174-01 9.420231+00	-5.386086-03 -4.295301-06	-4.930277-02 3.061891-07	-4.364002-02	7.228277-03	0.000000	0.000000	8.443279-02	0.000000
.180	1.553593+00 9.420176+00	-4.025128-03 -7.603879-06	2.662627-02 4.982056-07	-3.618107-02	1.093174-02	0.000000	0.000000	9.048270-02	0.000000
.200	1.459445+00 9.420087+00	1.502211-03 -1.200220-05	5.838693-02 7.863990-07	-1.437674-02	7.519063-03	0.000000	0.000000	9.510565-02	0.000000

Figure 5.1c

CV

	1	2	3	4	5	6	7	8	9	10
1	1.4485+002	-2.0032+000	3.0569+000	9.0663-001	6.5815-001	0.0000	-4.3285-001	4.6651+000	-2.5971-001	-6.1640+001
2	-2.0032+000	4.2781-002	-4.2971-002	-3.2405-002	-1.5078-002	0.0000	1.6735-002	-4.0396-002	1.0041-002	3.1969+000
3	3.0569+000	-4.2971-002	2.6351-001	8.8279-002	2.1298-003	0.0000	5.5980-002	9.1819-002	3.3588-002	1.0260+000
4	9.0663-001	-3.2405-002	8.8279-002	3.0847-001	-2.7849-002	0.0000	3.0505-002	-8.0203-002	1.8303-002	1.0612+001
5	6.5815-001	-1.5078-002	2.1298-003	-2.7849-002	2.8120-002	0.0000	-1.5659-002	2.0073-002	-9.3953-003	-3.0856+000
6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7	-4.3285-001	1.6735-002	5.5980-002	3.0505-002	-1.5659-002	0.0000	6.2500-002	0.0000	3.7500-002	-6.4651-003
8	4.6651+000	-4.0396-002	9.1819-002	-8.0203-002	2.0073-002	0.0000	0.0000	2.5000-001	0.0000	-1.0564-001
9	-2.5971-001	1.0041-002	3.3588-002	1.8303-002	-9.3953-003	0.0000	3.7500-002	0.0000	2.2500-002	-3.8791-003
10	-6.1840+001	3.1969+000	1.0260+000	1.0612+001	-3.0856+000	0.0000	-6.4651-003	-1.0564-001	-3.8791-003	9.0190+003
11	-7.5751-003	-2.4133-005	-8.0702-004	-2.3715-003	5.1715-004	0.0000	-6.9712-004	9.1693-005	-4.1827-004	-3.4956-001
12	7.2495-004	-1.0106-005	2.8314-004	4.4291-004	-6.4513-005	0.0000	1.1236-004	-7.2900-005	6.7416-005	8.3257-002
13	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

CV

	11	12	13
1	-7.5751-003	7.2495-004	0.0000
2	-2.4133-005	-1.0106-005	0.0000
3	-8.0702-004	2.8314-004	0.0000
4	-2.3715-003	4.4291-004	0.0000
5	5.1715-004	-6.4513-005	0.0000
6	0.0000	0.0000	0.0000
7	-6.9712-004	1.1236-004	0.0000
8	9.1693-005	-7.2900-005	0.0000
9	-4.1827-004	6.7416-005	0.0000
10	-3.4956-001	8.3257-002	0.0000
11	3.8727-005	-6.2630-006	0.0000
12	-6.2630-006	1.5167-006	0.0000
13	0.0000	0.0000	0.0000

NUMBER OF NON-ZERO CONTROLS IS 6
IU = 2 3 4 5 6 7 0 0

R

	1	2	3	4	5
1	7.1708-002	0.0000	0.0000	0.0000	0.0000
2	0.0000	2.1179-005	0.0000	0.0000	0.0000
3	0.0000	0.0000	1.3645-004	0.0000	0.0000
4	0.0000	0.0000	0.0000	1.5271-004	0.0000
5	0.0000	0.0000	0.0000	0.0000	1.3921-005

Figure 5.1d

The initial noise covariance matrix, R, is also printed in Figure 5.1(d). Because none of the instrument noise covariances were specified by the input deck, each of the diagonal elements of R was approximated as

$$R_{jj} = \frac{0.05}{N} \sum_{i=1}^{N} [y_j(t_i)]^2 \qquad (5.2)$$

That is, the variance of the noise in the jth instrument is approximated as 5% of the variance in the instrument's signal.

Information on the integration of the state and sensitivity differential equations for the zero-th iteration of the Newton-Raphson search begins in Figure 5.1e. A "matrix" of initial conditions appears in this figure. The first column is the initial condition of the $\hat{x}$ vector (estimated states). This column has length four because only four of the nine states are being integrated. The second column is the initial condition of $\partial\hat{x}/\partial\hat{\theta}_1$, i.e., the sensitivity of the estimated states to the first identified parameter, which is P(9) in the example. The third column is the initial condition of $\partial\hat{x}/\partial\hat{\theta}_2$, etc. The second and succeeding columns should always be zero unless the initial conditions of the states are among the identified parameters.

The four groups of values printed between the first line of dashes and the line of stars in Figure 5.1e are intermediate results in computing the time derivatives of $\hat{x}$ and $\partial\hat{x}/\partial\hat{\theta}$. First time (T) and the expansion variables (Z) at that time are printed. In the example, there are two expansion variables. Next is the u-vector showing the controls and look-up states (appended as additional controls) at time T. The four normal controls are zero. The first look-up state, $x(3) \equiv w$, equals 9.4202. The second

```
INITIAL CONDITIONS
          1            2            3            4
   1   0.0000       0.0000       0.0000       0.0000
   2   0.0000       0.0000       0.0000       0.0000
   3   0.0000       0.0000       0.0000       0.0000
   4   0.0000       0.0000       0.0000       0.0000
----------------------------------------------------------------------------------------------------------------------------
 T, Z =      0.0000       0.0000       0.0000
 U-VECTOR    0.0000       0.0000       0.0000       0.0000       9.4202+000   0.0000       0.0000       0.0000
 X-VECTOR    1.7975+002   0.0000       9.4202+000   0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 V,BVP,CBVO,OD,OS,OC,BVP,BVR           1.8000+002   0.0000       0.0000       3.8504+001   4.7286+001   1.0512+005   0.0000       0.0000
 C =         0.0000       0.0000       0.0000       0.0000       0.0000       0.0000      -4.2680-003  -8.8920-002   4.4470-001   3.1210-(
             3.4680-001   0.0000       0.0000       0.0000       0.0000       0.0000       0.0000      -5.3540-004  -5.5000-001   1.5420-(
             1.8520+001   2.7030+002   0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       5.5300-004  -6.1530-(
            -1.4930-001   5.9940-003  -1.3210-001
 DXI =       0.0000       0.0000       0.0000       0.0000
P( 9) SENSITIVITIES
 T, ZTH =    0.0000       0.0000       0.0000
 D(V,BVP,CBVO,OD,OS,OB,OC,BVR)/DP      0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 DC =        0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       1.0000+000   0.0000
             0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000
 DXTHI =     0.0000       0.0000       0.0000       0.0000
P( 20) SENSITIVITIES
 T, ZTH =    0.0000       0.0000       0.0000
 D(V,BVP,CBVO,OD,OS,OB,OC,BVR)/DP      0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 DC =        0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000
 DXTHI =     0.0000       0.0000       0.0000       0.0000
P( 21) SENSITIVITIES
 T, ZTH =    0.0000       0.0000       0.0000
 D(V,BVP,CBVO,OD,OS,OB,OC,BVR)/DP      0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 DC =        0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000
 DXTHI =     0.0000       0.0000       0.0000       0.0000
********************************************************************KOUNT =      0**************************************************
XHAT(   1)   0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
YHAT(   1)   0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
             0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
Y-VECTOR     1.1326-01   -4.3616-03   -3.5336-02   -8.8824-03    2.3608-03
----------------------------------------------------------------------------------------------------------------------------
 T, Z =      0.0000       0.0000       0.0000
 U-VECTOR    0.0000       0.0000       0.0000       0.0000       9.4202+000   0.0000       0.0000       0.0000
 X-VECTOR    1.7975+002   0.0000       9.4202+000   0.0000       0.0000       0.0000       0.0000       0.0000       0.0000
 V,BVP,CBVO,OD,OS,OC,BVP,BVR           1.8000+002   0.0000       0.0000       3.8504+001   4.7286+001   1.0512+005   0.0000       0.0000
 C =         0.0000       0.0000       0.0000       0.0000       0.0000       0.0000      -4.2680-003  -8.8920-002   4.4470-001   3.1210-0
             3.4680-001   0.0000       0.0000       0.0000       0.0000       0.0000       0.0000      -5.3540-004  -5.5000-001   1.5420-0
```

Figure 5.1e

and third look-up states, $x(5) \equiv q$ and $x(8) \equiv \theta$, are zero. A potential fourth look-up state is printed, even though it is not defined.

The x-vector printed next is the nine-element state vector. For example, $x(1) \equiv u = 179.75$ and $x(3) \equiv w = 9.4202$. Next are printed a series of terms which appear in the equations of motion.

$$
\begin{aligned}
V &= V_\infty \\
BVP &= \frac{b}{2V_\infty} p \\
CBVQ &= \frac{\bar{c}}{2V_\infty} q \\
QD &= \frac{1}{2} \rho V_\infty^2 = q_\infty \\
QS &= \frac{q_\infty S}{m} \\
QB &= q_\infty S b \\
QC &= q_\infty S \bar{c} \\
BVR &= \frac{b}{2V_\infty} r
\end{aligned}
\tag{5.3}
$$

The 33 aerodynamic coefficients (C) are printed in order across the page; hence, $C(11) \equiv C_{y_{\delta r}} = 0.3468$. Last in this group appear the time derivatives of the state variables being integrated, DXI. In this example, there are four: $\dot{v}$, $\dot{p}$, $\dot{r}$, and $\dot{\phi}$.

The second group contains information on sensitivities with respect to the first identified parameter, which is $P(9) \equiv C_{\ell_p}$. Time is repeated and followed by ZTH, the partial derivatives of expansion variables with respect to P(9). Next are the partial derivatives of the quantities defined in Eqs. (5.3). The partial

derivatives of the aero coefficients are "DC." Note that DC(19) = 1 because $C(19) = C_{\ell_p} = P(9)$. Last in this group is DXTHI, which is $d[\partial\hat{x}/\partial P(9)]/dt$ for the four integrated state variables.

Groups three and four are similar to group two except that all partial derivatives are with respect to P(20) and P(21), the second and third identified parameters, respectively. Note specifically that at T = 0, DC(17) = 0 in both groups three and four, even though

$$C(17) = P(7) + P(20)(Z_2 - Z_{2_0})^1 + P(21)(Z_2 - Z_{2_0})^3$$

$$= P(7) + P(20)\delta_{sp} + P(21)\delta_{sp}^3 \qquad (5.4)$$

This is because the partial derivative of C(17) with respect to either P(20) or P(21) is zero when $\delta_{sp} = 0$, which it does here.

The information printed between the line of stars and the second line of dashes in Figure 5.1e shows the results of the integration at the j-th sample point. In this instance, j = 1.

In the middle of the line of stars in Figure 5.1e is printed KOUNT, which is a running total of the number of times subroutine STEP has been called. This is approximately the number of times the states' and sensitivities' time derivatives have been computed. More will be said of its importance in Chapter VI.

Next is "XHAT(1)" which is the vector of integrated states and their sensitivities with respect to the identified parameters at the (in this case) first sample point ($t_1 = 0$). They appear in the order

$$\hat{x}_1,\ \hat{x}_2,\ \hat{x}_3,\ \hat{x}_4,\ \partial\hat{x}_1/\partial\theta_1,\ \partial\hat{x}_2/\partial\theta_1,\ \partial\hat{x}_3/\partial\theta_1,\ \partial\hat{x}_4/\partial\theta_1,$$

$$\partial\hat{x}_1/\partial\theta_2,\ \ldots,\ \partial\hat{x}_4/\partial\theta_3$$

They are followed by "YHAT", the measurement estimates and their sensitivities with respect to the identified parameters in similar order. "Y-VECTOR" is the vector of actual measurements at the same time.

As the integration proceeds, the groups shown in Figure 5.1e are repeated at intervals specified by the program's input. These pages are omitted from the figure to save space. Figure 5.1f has the very end of this stream of information at its top.

The RCMP matrix is printed next. Its diagonal elements are computed as

$$RCMP_{i,i} = \frac{1}{N} \sum_{j=1}^{N} [y_i(t_j) - \hat{y}_i(t_j)]^2 \tag{5.5}$$

These are the variances of the errors between the actual and the estimated measurements; and, therefore, are a measure of the goodness of the fit. Also, it is the best estimate of the measurement noise covariance (see Appendix A).

Figure 5.1g presents the results of the zero-th (i.e., initial) iteration of the Newton-Raphson search. The DJ vector is the partial of J, the negative log-likelihood cost function, with respect to the three identified parameters. The D2J matrix is the $\partial^2 J/\partial\theta_j \partial\theta_k$ matrix. Also printed are its inverse, its eigenvalues, and its eigenvectors. The significance of the number of "eigenvalues effecting a full-sized step" will be deferred to Chapter VI.

Next are printed the values of the parameter estimates, the lower bounds on the error standard deviations of these estimated values, and their F values, which are the ratios of the squares of the parameter values to the squares of their respective standard deviations. The value of the likelihood function and the length of the gradient are printed. Of course, it is the likelihood function that the program is seeking to minimize, and the length of the gradient will be zero at a minimum.

```
                    0.0000        0.0000        0.0000
   DXTHT =      3.5164-003 -4.5561-004  7.2415-006  5.5040-005
**********************************************************KOUNT =    55*********************************************************
XHAT( 101)   -4.3999+000 -2.0047+002  5.8448-003  1.0937-003  4.3346-001 -3.8806-002 -1.1113-002  1.0676-002  2.0126+000 -1.6165-0
              1.8513-002  1.7133-002  1.2234-003 -8.3617-005  1.1918-005  8.4559-006
YHAT( 101)    9.2539+001 -2.4440+002 -2.0047+002  5.8448-003  1.0937-003 -5.5298-002  1.3237-003 -2.1343-002 -6.1123-003  5.9820-0
             -2.1523-001  1.1175-002 -1.6165-001  1.8513-002  1.7133-002 -1.3982-004  6.7929-006 -8.3617-005  1.1918-005  8.4559-0
Y-VECTOR        1.1577+00   -1.7674-02  -2.8158-02  -1.6233-03  -1.1160-03

RCMP
             1             2             3             4             5
     1   9.1414-002    0.0000        0.0000        0.0000        0.0000
     2   0.0000        3.2083-005    0.0000        0.0000        0.0000
     3   0.0000        0.0000        6.5414-004    0.0000        0.0000
     4   0.0000        0.0000        0.0000        2.3160-004    0.0000
     5   0.0000        0.0000        0.0000        0.0000        1.5810-004

SUBROUTINE STEP  WAS CALLED   55 TIMES.
SUBROUTINE COFF  WAS CALLED  118 TIMES.
SUBROUTINE DCOEF WAS CALLED  354 TIMES.
```

Figure 5.1f

```
LATERAL CRUISE DYNAMICS EXAMPLE                                  *NLSCIDNT PROGRAM*

*********************
* ITERATION NO.  0 *
*********************


DJ....GRADIENT
                1
      1    4.2990+000
      2   -5.1966+002
      3   -3.2734-001


D2J...INFORMATION MATRIX
                1              2              3
      1    1.4918+003   8.3240+002   6.0660-001
      2    8.3240+002   1.9155+004   1.2292+001
      3    6.0660-001   1.2292+001   8.0144-003


D2JI...INVERSE OF INFORMATION MATRIX
                1              2              3
      1    7.0715-004   2.2894-004  -4.0466-001
      2    2.2894-004   3.3795-003  -5.2005+000
      3   -4.0466-001  -5.2005+000   8.1316+003

EIGENVALUES=  1.91941+004 1.45261+003 1.22977-004


EIGENVECTORS OF INFO MATRIX INVERSE
                1              2              3
      1    4.6970-002  -9.9890-001   4.9764-005
      2    9.9890-001   4.6970-002   6.3954-004
      3    6.4118-004  -1.9670-005  -1.0000+000

EIGENVALUES EFFECTING A FULL-SIZED STEP =  3


 NO   NAME    PARAMETER VALUE   STD DEV OF P     F-VALUE
 --   ------  ---------------   --------------   --------------
 19   CLP      -5.5000000-001   1.4625742-002    1.4141307+003
 20   CLDS      0.0000000       5.8133050-002    2.9590601+002
 21   CLDS3     0.0000000       9.0175226+001    1.2297746-004

 LIKELIHOOD FUNCTION =  -1092.49

 LENGTH OF GRADIENT  =     519.68
```

Figure 5.1g

At the top of Figure 5.1h are the "new parameter values" found by adding the "step" to the "parameter values" printed earlier. These new parameter values are presumed to result in a value of the likelihood function less than the one obtained for this iteration, and they are the parameter values which will be used at the beginning of the next iteration. "Normalized step" is the ratio of the step vector to the parameter values in Figure 5.1g, except that division by a zero parameter value is replaced by division by one. This ratio tells at a glance the relative size of the step, and it will decrease as a minimum is approached.

R is the measurement noise covariance which will be used in the next iteration. What follows it is actually part of the next iteration and parallels the printout of Figure 5.1e.

This example run was allowed only one iteration and, therefore, the parameter search did not converge. Thus, the message "maximum number of iterations exceeded" appears in Figure 5.1j. The matrix A is the sum of the a priori information matrix, M_{ap}, and the information matrix from the last iteration, M_{last}. The vector B is the sum

$$B = M_{ap} P_o + M_{last} P_{last} \tag{5.6}$$

where

P_o = the initial (input) values of the identified parameters, and

P_{last} = the values of these parameters used in the last iteration (not the last "new parameter values")

Finally,

$$PSTAR = A^{-1}B = (M_{ap}+M_{last})^{-1}(M_{ap}P_o + M_{last}P_{last}) \tag{5.7}$$

```
NO   NAME     NEW PARAM VALUE       STEP           NORMALIZED STEP
--   ------   ---------------   ---------------   ---------------
 9   CLP       -5.5909034-001   -9.0903012-003    -1.6527820-002
20   CLDS       5.2865576-002    5.2865576-002     5.2865576-002
21   CLDS3     -3.8986584+001   -3.8986584+001    -3.8986584+001


R
   -        1           2           3           4           5
   1    7.1708-002  0.0000      0.0000      0.0000      0.0000
   2    0.0000      2.1179-005  0.0000      0.0000      0.0000
   3    0.0000      0.0000      1.3045-004  0.0000      0.0000
   4    0.0000      0.0000      0.0000      1.5271-004  0.0000
   5    0.0000      0.0000      0.0000      0.0000      1.3921-005

INITIAL CONDITIONS
   -        1           2           3           4
   1    0.0000      0.0000      0.0000      0.0000
   2    0.0000      0.0000      0.0000      0.0000
   3    0.0000      0.0000      0.0000      0.0000
   4    0.0000      0.0000      0.0000      0.0000
------------------------------------------------------------------------------------------------------------------------
  T, Z =     0.0000      0.0000      0.0000


  U-VECTOR   0.0000      0.0000      0.0000      0.0000      9.4202+000  0.0000      0.0000
  X-VECTOR   1.7975+002  0.0000      9.4202+000  0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
  V,BVP,CBVD,DD,DS,DC,BVP,BVR         1.8000+002  0.0000      0.0000      3.8504+001  4.7286+001  1.0512+005  0.0000      0.0000
  C =        0.0000      0.0000      0.0000      0.0000      0.0000      0.0000     -4.2680-003 -8.8920-002  4.4470-001  3.1210-0(
             3.4680-001  0.0000      0.0000      0.0000      0.0000      0.0000      0.0000     -5.3540-004 -5.5909-001  1.5420-0(
             1.8520-001  2.7030-002  0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      5.5300-004 -6.1530-0(
            -1.4930-001  5.9940-003 -1.3210-001
  DXI =      0.0000      0.0000      0.0000      0.0000
P( 9) SENSITIVITIES
  T, ZTH =   0.0000      0.0000      0.0000
  D(V,BVP,CBVD,DD,DS,DB,DC,BVR)/DP   0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
  DC =       0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
             0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      1.0000+000  0.0000
             0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
             0.0000      0.0000      0.0000
  DXTHI =    0.0000      0.0000      0.0000      0.0000
P( 20) SENSITIVITIES
  T, ZTH =   0.0000      0.0000      0.0000
  D(V,BVP,CBVD,DD,DS,DB,DC,BVR)/DP   0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
  DC =       0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
             0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
             0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
             0.0000      0.0000      0.0000
  DXTHI =    0.0000      0.0000      0.0000      0.0000
P( 21) SENSITIVITIES
  T, ZTH =   0.0000      0.0000      0.0000
```

Figure 5.1h

```
*********************
* ITERATION NO.  1 *
*********************

DJ...GRADIENT
              1
     1    -5.9246-001
     2    -3.4605-001
     3    -5.9880-002

D2J...INFORMATION MATRIX
              1            2            3
     1     1.5644+003   6.5034+001   3.3945+001
     2     6.5034+001   5.2342+001   2.4769+001
     3     3.3945+001   2.4769+001   1.1897+001

D2JI...INVERSE OF INFORMATION MATRIX
              1            2            3
     1     7.0099-004   5.0960-003  -1.2610-002
     2     5.0960-003   1.3266+000  -2.7765+000
     3    -1.2610-002  -2.7765+000   5.9004+000

EIGENVALUES=  1.56796+003 6.05217+001 1.38684-001

EIGENVECTORS OF INFO MATRIX INVERSE
              1            2            3
     1     9.9881-001  -4.8684-002   1.8833-003
     2     4.3225-002   9.0330-001   4.2675-001
     3     2.2477-002   4.2616-001  -9.0437-001

EIGENVALUES EFFECTING A FULL-SIZED STEP =  3
```

NO	NAME	PARAMETER VALUE	STD DEV OF P	F-VALUE
9	CLP	-5.5909030-001	1.4802610-002	1.4265511+003
20	CLDS	5.2865576-002	6.0890663-002	7.5377986-001
21	CLDS3	-3.8986580+001	9.4701543+001	1.6947914-001

LIKELIHOOD FUNCTION = -1099.94

LENGTH OF GRADIENT =   .69

Figure 5.1i

08/02/76 17:42:40 MOHR 0373AAF95 000373 S58 200 DATE 080276

LATERAL CRUISE DYNAMICS EXAMPLE *NLSCIDNT PROGRAM*

NO	NAME	NEW PARAM VALUE	STEP	NORMALIZED STEP
19	CLP	-5.5829431-001	7.9598774-004	1.4237194-003
20	CLDS	6.8505661-002	1.5690085-002	2.9584631-001
21	CLDS3	-6.2961505+001	-2.3974921+001	-6.1495311-001

*** MAXIMUM NUMBER OF ITERATIONS EXCEEDED.

A--NEW INFORMATION MATRIX

	1	2	3
1	9.3268+003	2.0841+003	8.6596+001
2	2.0841+003	6.9316+004	2.0137+002
3	8.6596+001	2.0137+002	1.2369+001

B

	1
1	-6.4639+003
2	-2.1097+003
3	-5.1046+002

PSTAR--NEW PARAMETERS

	1
1	-3.3897-001
2	9.7350-002
3	-4.0479+001

Figure 5.1j

which is the vector of the optimal parameter estimates which combine the information obtained from this run and the information known a priori. Of course, to be valid this run should have been allowed to converge. A, B, and PSTAR will not be printed if the a priori information matrix is zero.

Next are printed plots of the actual and estimated measurement time histories and the control time histories. Figure 5.1k is an example of the measurement plots and Figure 5.1ℓ of the control plots. In the measurement plots, "A" is always printed for the actual measurement and "B" for the estimated measurement. Control variables are plotted two to a graph; and, again, look-up states are treated as additional controls.

5.1.2 Printout Control

In most cases, the detail of the printout is controlled by the flag IPRNT in card type 3 of Table 4.1. For normal production runs this flag is set to zero, and the printout is shortened to essential results of each Newton-Raphson iteration. Setting IPRNT = 1, 2, or 3 results in increasingly greater detail of diagnostic information, as explained in Table 5.1.

Setting IPRNT = 1 results mainly in added information regarding the computation of the parameter step in each iteration. Setting IPRNT = 2 results mainly in the printing of $\hat{x}$, $\partial\hat{x}/\partial\theta$, $\hat{y}$, and $\partial\hat{y}/\partial\theta$ as the trajectory is integrated for each iteration. Finally, setting IPRNT = 3 results in detailed information on the trajectory integration.

5.2 MASS STORAGE OUTPUT

An option is provided to write on mass storage: (a) the last vector of identified parameters, (b) the last information matrix,

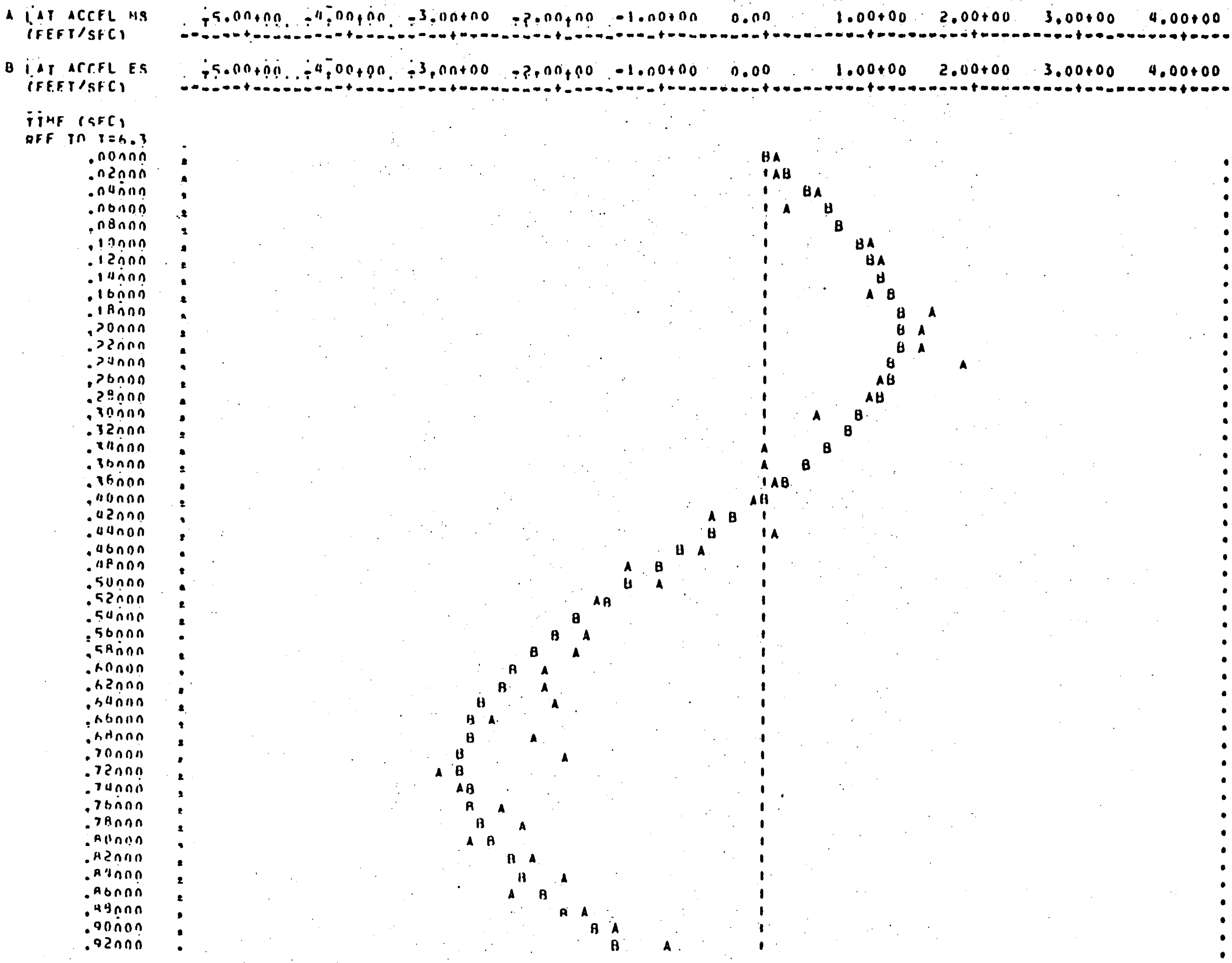
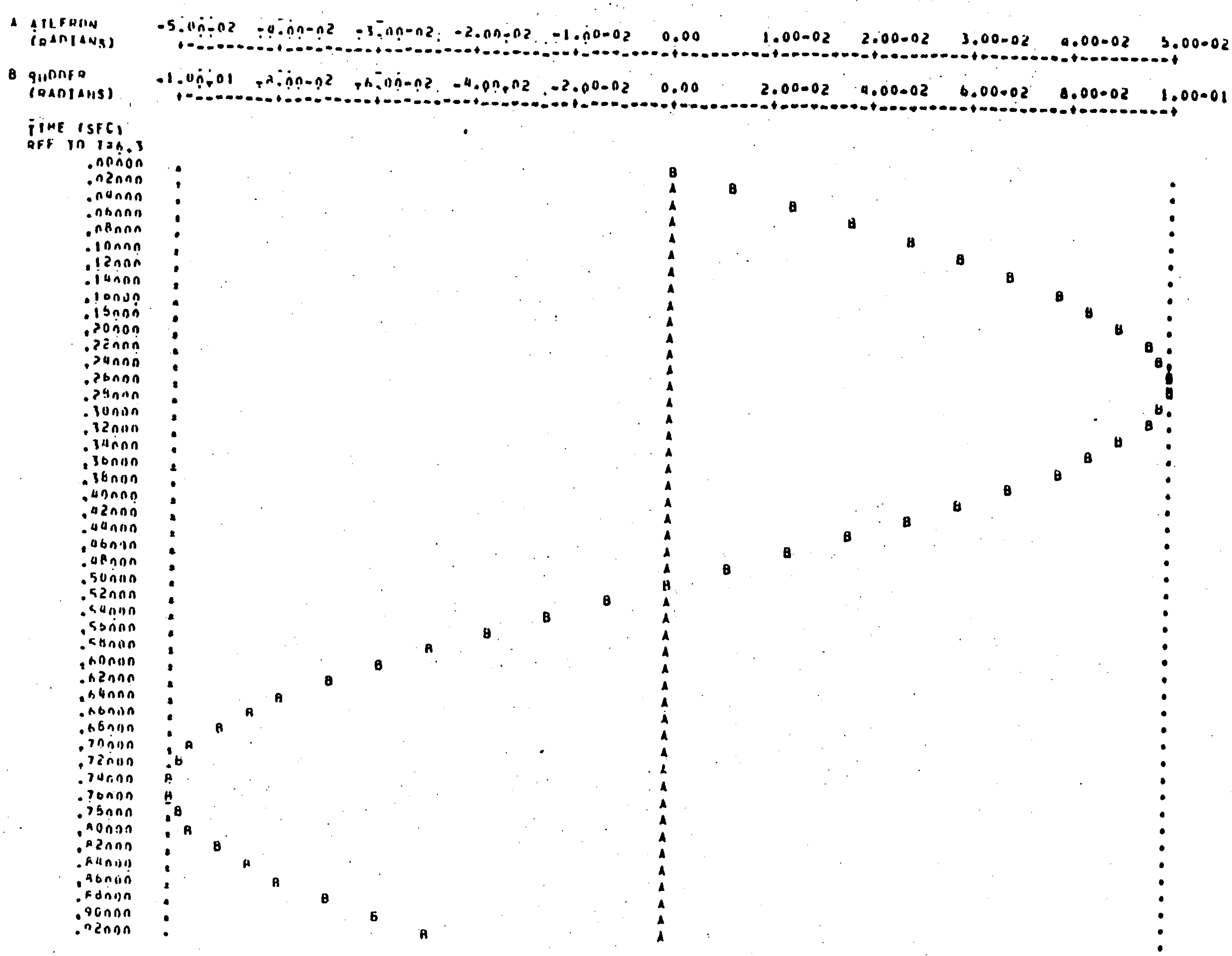

A LAT ACCEL HS (FEET/SEC)
B LAT ACCEL ES (FEET/SEC)
-5.00+00 -4.00+00 -3.00+00 -2.00+00 -1.00+00 0.00 1.00+00 2.00+00 3.00+00 4.00+00
TIME (SEC) REF TO T=6.3

Figure 5.1k



A AILERON (RADIANS)
-5.00-02 -4.00-02 -3.00-02 -2.00-02 -1.00-02 0.00 1.00-02 2.00-02 3.00-02 4.00-02 5.00-02
B RUDDER (RADIANS)
-1.00-01 -8.00-02 -6.00-02 -4.00-02 -2.00-02 0.00 2.00-02 4.00-02 6.00-02 8.00-02 1.00-01
TIME (SEC) REF TO 136.3

Figure 5.1l

Table 5.1

Printed Information Controlled by IPRNT Flag

FLAG VALUE	INFORMATION PRINTED	COMMENTS
IPRNT = 0	1. Card input data repeated	Only parameters mentioned in the input deck are printed under "Initial Parameter Values." The a priori information matrix is printed only if it is non-zero.
	2. For each iteration of the Newton-Raphson search: a. Parameter values at start of iteration b. Parameters' standard deviations c. Parameters' F-values d. Value of negative log likelihood function, J e. Length of first gradient of J f. New parameter estimates g. Parameter steps h. Innovation covariance (RCMP)	
	3. Final combined parameter set (PSTAR)	Only if a priori information is non-zero
IPRNT = 1	Same as for IPRNT = 0 plus:	
	1. Values in ISFLAG, NSFLAG, NMFLAG, NZFLAG, and IU arrays	These arrays described in Section 5.1.1
	2. CV matrix	Described in Section 5.1.1
	3. Initial R matrix	
	4. For each iteration of the Newton-Raphson search: a. First gradient of J (DJ) b. Second gradient of J (D2J) c. Eigenvalues and eigenvectors of $(D2J)^{-1}$ and number of eigenvalues effecting a full-sized step	Significance described in Chapter VI
	5. Combined information matrix (A)	Only if a priori information matrix is non-zero.

Table 5.1 (Continued)

FLAG VALUE	INFORMATION PRINTED	COMMENTS
IPRNT = 2	Same as for IPRNT = 1 plus for each iteration of the Newton-Raphson search: a. States and their sensitivity estimates (XHAT) b. Measurements and their sensitivity estimates (YHAT) c. Actual measurements (Y-VECTOR) d. Number of calls to subroutines STEP, CØEF, AND DCØEF e. Inverse of D2J matrix	Printed for the first 4 sample points and for every (INCPR2)-th sample point Significance described in Section VI.
IPRNT = 3	Same as for IPRNT = 2 plus for each iteration of the Newton-Raphson search: a. Initial conditions of states and their sensitivities b. Time and values of expansion variables (T and Z) c. Control vector including look-up states (U-VECTOR) d. Nine-element state vector (X-VECTOR) e. Values of V, BVP, CBVQ, QD, QS, QC, BVP, BVR f. Aero coefficients (C) g. Time derivatives of integrated states (DXI) h. Partial derivatives of variables in b through g above with respect to the identified parameters	Printed for the first 3 sample points and for every (INCPR3)-th sample point

(c) the time histories of the actual and estimated measurements, and (d) the time histories of the control variables. The mass storage device (tape drive or disk) is currently designated as logical unit 3. The data are written in unformatted records by statements equivalent to the following:

```
      WRITE(3) NS,NQ,NP,NN,M,(PID(J),J=1,M),((INFØL(J,K),
     1   J=1,M),K=1,M)
      DØ 10 K=1,NN
   10 WRITE(3) T(K),(U(J,K),J=1,NQ),(Y(J,K),J=1,NP),
     1   (YHAT(J,K),J=1,NP)
      ENDFILE 3
```

where

NS = number of states integrated in the run,

NQ = number of control variables in the model plus number of possible look-up states (currently NQ=8)

NP = number of measurements used in the run,

NN = number of sample points in the run,

M = number of parameters identified in the run,

PID = vector of identified parameters used in the last iteration of the run,

INFØL = last information matrix,

T = array of time values at sample points,

U = array of control and look-up state variables at sample points,

Y = array of actual measurements at sample points, and

YHAT = array of estimated measurements at sample points.

CHAPTER VI

EFFECTIVE USE OF THE NLSCIDNT PROGRAM

This chapter contains information on selected topics and guidelines for effective use of the NLSCIDNT Program.

6.1 INTEGRATION ALGORITHM

Each outer iteration of the Levenberg-Marquardt search requires the integration of $n(m+1)$ ordinary differential equations, where n is the number of integrated states and m is the number of parameters identified. This is the most time-consuming task the program must perform. Most of this time is spent evaluating $d\hat{x}/dt$ and $d(\partial\hat{x}/\partial\theta)/dt$. Therefore, an integration algorithm which requires relatively few evaluations of these derivatives is more efficient than one requiring more.

Adams formulas are well known for this desirable characteristic, but most implementations are not self-starting (for example, four Runge-Kutta steps may have to be taken first), they have a fixed order (usually they require four previous points), and they have clumsy machinery for varying the stepsize (which may cost them much of their inherent advantage in speed). A recent implementation of an Adams formulation due to Shampine and Gordon [3] overcomes all these objections. It is called a PECE method because for each step it:

- predicts the solution of the differential equations at the end of the step using the Adams-Bashforth predictor of order k,

- evaluates the derivative at this predicted solution point,

- corrects its solution value at the end of the step using the Adams-Moulton corrector of order $k+1$, and

- evaluates the derivative again using the corrected solution.

The PECE method was designed to be self-starting. Also, the order and the stepsize are both varied routinely to reduce the number of evaluations of the derivatives while meeting relative and absolute error bounds imposed on the solution by the user. These bounds are imposed as follows. Let $\hat{\gamma}_i$ be any element of the $\hat{x}$ and $\partial\hat{x}/\partial\theta$ vectors as computed by the PECE method at the i-th point, and let ε_i be the difference between $\hat{\gamma}_i$ and its true value, γ_i, at that point. Because γ_i is unknown, ε_i can only be estimated. However, the theory says that η_i, the difference between the predicted and the corrected values for $\hat{\gamma}_i$, is an upper bound on the error ε_i. That is,

$$|\varepsilon_i| \leq |\eta_i| \tag{6.1}$$

The program bounds the error by requiring

$$|\eta_i| \leq \text{RELERR}*|\hat{\gamma}_i| + \text{ABSERR} \tag{6.2}$$

where RELERR and ABSERR are inputs to the program. If the user does not supply values, they default to 10^{-5} and 1.7×10^{-4}, respectively. (The value 1.7×10^{-4} is 0.01 degrees expressed in radians.) Note that choosing ABSERR=0 will cause difficulties if the solution $\hat{\gamma}_i$ passes through zero.

If the error bounds are unnecessarily restrictive, then the program will be forced to take many small steps and waste time. If the error bounds are too lax, the solution of the differential equations will be inaccurate, and the subsequently computed step in the identified parameters will also be in error. The user should experiment to find values of RELERR and ABSERR which are best suited to the problems he solves.

The program constrains the order of the predictor equation between 1 and 12. That is, as few as one and as many as twelve

past time points of data may be used to predict the next. Obviously, this requires setting aside sufficient storage for the maximum of twelve past points, which is $12n(m+1)$ locations. Thus, higher orders quickly become impractical and produce diminishing returns anyway. In practice, twelth-order is seldom reached.

The maximum permitted integration step-size is ten times the data sample step-size. The minimum is 0.02 times the sample step-size, unless another factor is read into the program (MAXNM1 in Table 4.1). In general, the ratio of the integration step-size to the sample step-size or the inverse ratio is not an integer. Hence, the "mesh" points of the integration (i.e., the time points which mark the end of the integration steps taken) almost never coincide with data sample times. The values of $\hat{x}$ and $\partial\hat{x}/\partial\theta$ must be found by interpolation between their values at mesh points. This is an inexpensive task, however, since the interpolating polynomial is readily available from the correction equation. Furthermore, the theory says that this interpolating polynomial is as accurate between mesh points as at them.

In a typical run, the integration step-size is initially much smaller than the sample step-size. In fact, it may be repeatedly halved and the time derivatives of $\hat{x}$ and $\partial\hat{x}/\partial\theta$ repeatedly computed until a small enough first step is taken to satisfy the error bounds. Once started, however, the integration step-size is steadily increased until it is 3 to 7 times the sample step size. Consequently, the run time is not directly proportional to the number of data points (see Figure 6.1).

In the example run whose printed output is shown in Figure 5.1, the initial number of integration steps per sample step was 6.0. By the 100th sample point, this ratio was down to 0.32. The total number of steps taken is the number of times subroutine STEP is called. Because STEP may occasionally make more than one

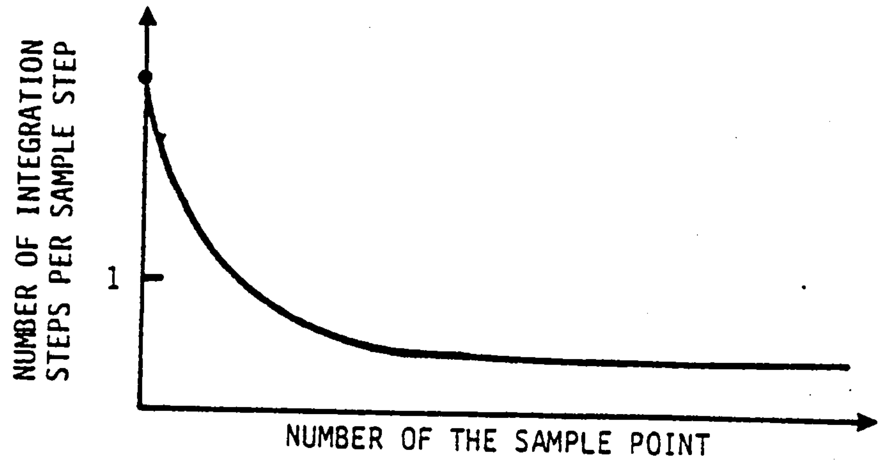


Figure 6.1 Variation of Integration Step-Size in a Typical Run

attempt at a successful step in order to satisfy the error bounds, the actual total number of times $d\hat{x}/dt$ and $d(\partial\hat{x}/\partial\theta)/dt$ are evaluated may be slightly higher than twice the number of steps. (If STEP made no "false starts," it would be exactly twice.) The number of times subroutine CØEF is called reflects the number of evaluations of $d\hat{x}/dt$. Because subroutine DCØEF is called once in evaluating $d(\partial\hat{x}/\partial\theta_i)/dt$ for each θ_i, $i=1,\ldots,m$, it will be called m times more often than CØEF.

The reader may now appreciate the efficiency of the PECE algorithm. In 101 sample points, it needed to evaluate the time derivatives only 118 times. The second-order Runge-Kutta algorithm would have required 202 evaluations of the time derivatives, for example. Greater savings are realized for longer data records.

The cause and possible solution to error conditions in the use of subroutine STEP will be discussed briefly in Chapter VII For

more detailed information on the source of error conditions as well as more details on the theory and use of the PECE algorithm, the reader is referred to Ref. 3.

6.2 CONVERGING TO A MINIMUM (AND OTHERWISE)

6.2.1 Convergence Criteria

After each Levenberg-Marquardt iteration, the program performs two tests to determine if it has found a minimum of the cost function. If either test gives a positive result, the search terminates. These tests are:

(1) $$\left|\frac{J_k - J_{k-1}}{J_{k-1}}\right| < 10^{-5} \tag{6.3}$$

where J_k is the cost after the current iteration and J_{k-1} is the cost after the previous outer iteration.

(2) $$\frac{1}{m} \sqrt{(\Delta\theta)_N^T \, (\Delta\theta)_N} < 10^{-3} \tag{6.4}$$

where $(\Delta\theta)_N$ is the "normalized step" vector (explained in Section 5.1.1), and m is the number of identified parameters.

6.2.2 Step Cutting Procedure

These criteria have served the purpose well in all cases except one--step cuts. If the value of the cost function for an iteration is greater than the cost at the previous iteration, the program cuts back on the parameter step it has taken until a lower cost is found. Usually, this works satisfactorily, and the program

proceeds. However, sometimes round-off or other inaccuracies (such as error bounds which are too lax) cause the step to have been computed erroneously in the first place, and no amount of step cutting will result in a lower cost. The result is that the step is cut so much that one of the convergence criteria is met even though the cost may be nowhere near a minimum. The user should suspect this condition whenever he sees a large number of step cuts occurring just before the search appears to converge. A suggested remedy is given in Chapter VII.

Nonetheless, the step cutting procedure in NLSCIDNT has proved itself very effective. It is a common occurrence that the cost function may be nearly insensitive to one or more linear combinations of the identified parameters. In such cases, the parameter step has large components in these directions of the parameter space. In Appendix A it is shown that the parameter step vector is the sum of vectors whose directions are the same as the eigenvectors of $(\partial^2 J/\partial\theta^2)^{-1}$, i.e., the inverse of the information matrix, and whose lengths are inversely proportional to the eigenvalues of the same matrix. Therefore, a small eigenvalue corresponds to a large step component in the direction of the associated eigenvector.

The NLSCIDNT step-cutting procedure works as follows:

(1) set $k = 0$;

(2) if the current cost is less than the previous Newton-Raphson iteration's cost, proceed to the next iteration; otherwise,

(3) set $j = k + \text{integer}\,[1 + m/5]$, where m is the number of identified parameters, then set $k = j$;

(4) reduce the length of the components of the step vector in the directions associated with the k smallest eigenvalues and recompute the step vector.

(5) add this new step vector to the previous Newton-Raphson iteration's parameter vector and obtain a new parameter vector, compute its associated cost, and go back to step 2.

If step cuts have occurred previously $(k > 0)$ and a Newton-Raphson iteration is completed successfully without resorting to more step cuts, then k is reduced by one. When the lowest level of diagnostic printout is flagged, a message is printed to inform the user how many eigenvalues are permitted their full size in computing the step vector (see Figure 5.1g).

6.2.3 Primary and Secondary Parameters

The computer program can identify any or all unknown parameters if enough information is available about these parameters from the data. To identify a large number of parameters, a very careful procedure is often required to ensure convergence. In the procedure which has been found to be most successful, the parameters are divided into two or more groups in decreasing order of the effect they have on the rotorcraft response. Initially, only the most important parameters are identified, leaving the remaining ones fixed at a priori values. Once a reasonable convergence is achieved on these parameters, the second group may be added to the list of identified parameters. The identification is carried out using this new set of parameters, which are identified until a reasonable convergence is reached. This procedure is repeated until all the parameters are included in the identified set.

6.2.4 Initial Parameter Estimates

Good a priori values (e.g., from other wind tunnel data or flight tests) should be used for start-up. If they are not available, it may be necessary to use a least-squares type of procedure to obtain starting values. If the start-up values are too far from the true answer, the program may converge to a relative minimum away from the global minimum of the cost function.

6.2.5 Data Record Length

Sufficient data length should be used to identify parameters. If the data length is too short, the identified model may yield a good time history match even though the parameter estimates are inaccurate. A data length equal to 2 to 3 times the longest modal period of the system should prove adequate.

6.2.6 Data Sampling Interval

In order to assure adequate information content in the data, a sampling interval of at least 25 times the highest system natural frequency is required. A faster sampling rate provides somewhat more accurate parameter estimates because of some additional information available in the sampled data. However, the algorithm realizes diminishing returns in terms of increased parameter accuracy as the sampling rate approaches the continuous time case.

6.3 A PRIORI INFORMATION

Suppose a user has three data records at the same flight condition. Because the control input time histories differ among the three records, the information contents of the three records are different. By processing these records individually, the user will get three slightly different sets of parameter estimates and three sets of standard deviations. But what he wants is a single set which combines the information content of all three records.

When he has processed the first record and is about to process the second, he has a priori information from the first record about the parameters he is about to identify in the second record. He can combine the information of the two records by reading into the second NLSCIDNT run: (a) the converged para-

meter estimates of the first run as initial parameter estimates in the second, and (b) the converged information matrix of the first run as the a priori information matrix of the second. After the second run has converged, it uses this a priori information with the second record's information to produce a statistically combined final information matrix and parameter estimates according to Eqs. (5.6) and (5.7).

Similarly, when he processes the third record, he should read the combined parameter estimates and information matrix into the third run as the initial parameter estimates and a priori information matrix. After this run converges, the statistical combining is again performed, and the user obtains his goal--one set of parameter estimates and standard deviations combining the information of all three data records.

The user must take care that each run identifies exactly the same parameters in exactly the same order. The program has no way of checking whether this is true. If it is not true, the results are invalid.

The user may find it more convenient to perform this statistical combining of information himself after processing each of the three records individually. In this case, he must write a program to do the following:

$$M = \sum_{i=1}^{n} M_i \tag{6.3}$$

$$p^* = M^{-1} \sum_{i=1}^{n} M_i p_i \tag{6.4}$$

where

M_i is the converged information matrix from the i-th run,

p_i is the converged parameter estimates from the i-th run,

n is the number of runs whose answers are combined,

M is the final information matrix, and

p^* is the combined parameter estimates.

The M_i and p_i may be stored on tape by each run (see Section 5.2).

CHAPTER VII

DIAGNOSTICS

7.1 NORMAL TERMINATION PROBLEMS

Even though a run terminates normally by announcing it has converged, there may be problems with the result. Three problems are discussed below.

7.1.1 Large Number of Step-Cuts

If there were several step-cuts taken just before the search "converged," it may not actually be near a minimum, as explained in Section 6.2.2. Corrective action:

(1) Delete parameters having low identifiability from the search.

(2) Restart the program using the parameter values from the final iteration of the terminated run.

Increasing the step-cut limit (maximum number of inner iterations permitted) beyond 4 or 5 is generally not helpful but may be tried.

7.1.2 High Value for a Measurement Noise's Variance

The diagonal elements of the final RCMP matrix are estimates of the measurement noises' variances. If one or more are unreasonably large, a problem usually exists. Corrective action:

(1) Inspect the time history data for errors, particularly for disagreements with the program's sign conventions, for time lags (for air data errors, particularly), and for lost data due to a telemetry dropout.

(2) Inspect the initial parameter values, particularly for incorrect signs or units.

(3) Verify that the model equations are reasonable given the response data.

7.1.3 High Standard Deviation for a Parameter

If the standard deviation for a particular parameter is high, it implies that its identifiability is low. There are several possible causes:

(1) the parameter inherently has little effect on the aircraft's response,

(2) there is insufficient excitation of the mode for which that parameter is most influential,

(3) too many parameters are being identified relative to the information content of the data record, or

(4) the parameter's influence is masked by the parallel influence of a more important parameter.

Corrective action:

(1) Fix the parameter and identify more important parameters, then fix them at their final values and try again to identify the troublesome parameter.

(2) Try to find a data record having more information on the parameter and process it.

7.2 ABNORMAL TERMINATION PROBLEMS

If possible, an error message is printed when the program terminates abnormally. The conditions resulting in the error messages and recommended corrective actions (if not obvious) are discussed below.

*** ERRØR IN SUBRØUTINE INPUT. P(j) .NE. 0 BUT AN INCØRRECT CØEFFICIENT INDEX WAS SPECIFIED.

Parameters 1 through 400 are set aside for use in the polynomial expansions defining the aerodynamic coefficients. The j-th parameter was given a value but not assigned to the expansion of any aero coefficient; i.e., the input card describing the j-th parameter (card type 5 in Table 4.1) did not correctly specify the associated coefficient index number. Valid coefficient index numbers are in the range 1 to 384.

*** ERRØR IN SUBRØUTINE INPUT. A NØN-ZERØ EXPØNENT WAS DETECTED FØR AN UNDEFINED EXPANSIØN VARIABLE.

If only $n(\leq 5)$ expansion variables are defined for the run, then values for exponents should not appear for expansion variables $Z_{n+1},...,Z_5$ in parameter input cards (type 5 in Table 4.1). However, one was detected.

*** ERROR IN SUBRØUTINE INPUT. LTYPE = xx, WHICH IS NØT ØNE ØF THE ALLØWABLE SET:

Invalid characters were detected in the first two columns of a card of type 4 (Table 4.1).

*** ERRØR IN SUBRØUTINE FLAGIT. LL(j) = xx, WHICH IS NØT IN THE ALLØWABLE SET:

An ivalid input name of a state, measurement, or expansion variable was detected in an input card of type 4 (Table 4.1). Valid input names are listed in Table 4.2.

*** ERRØR IN SUBRØUTINE FLAGIT. THE NUMBER OF ELEMENTS SPECIFIED BY INPUT IS j, WHICH EXCEEDS THE MAXIMUM ALLØWED, k.

The number of variables specified in a card of type 4 (Table 4.1) exceeds the maximum number for which the program is dimensioned.

*** ERRØR IN SUBRØUTINE INREAD. DEVICE ERRØR DETECTED WHILE ATTEMPTING TØ READ DATA PØINT k. THE TIME AT THE PREVIØUS PØINT, T(K-1), WAS x.

Device errors usually result from selecting the wrong density or parity for a tape or attempting to read a damaged portion of the tape.

*** ERRØR IN SUBRØUTINE INREAD. END ØF FILE ENCØUNTERED WHILE ATTEMPTING TØ READ DATA PØINT k. THE TIME AT THE PREVIØUS PØINT, T(K-1), WAS x.

The number of data sample points specified in input card type 2 (Table 4.1) exceeds the number of logical records on the time history tape.

*** ERRØR IN SUBRØUTINE PRNTP. THE INDEX ØF THE CØEFFICIENT CØRRESPØNDING TØ PARAMETER (j) IS ZERØ.

Parameters 1 through 400 must appear in the polynomial expansion of an aero coefficient.

*** ERRØR IN SUBRØUTINE UPDATE. INVALID INPUT WAS PASSED TO SUBRØUTINE INTGR8.

The two most likely sources of invalid input to INTGR8 are:

(1) the maximum number of data points specified in card type 2 (Table 4.1) is $\leq$ 1, or

(1) restart the program using the parameter values from the final iteration of the terminated run, and

(2) increase the iteration limit.

*** MAXIMUM STEP-CUTS LIMIT, j, EXCEEDED. PLØTS, IF ANY, WILL SHØW Y-HAT FRØM THE PREVIØUS ITERATIØN.

The parameter set used at the beginning of the last Newton-Raphson iteration resulted in a higher cost than the previous iteration had. The program began cutting back the step in the parameters in search of a lower cost, but the number of step-cuts exceeded the limit specified in card type 2 (Table 4.1). Corrective action:

(1) Delete parameters with low identifiability (small F-values) from the search.

(2) Restart the program using the parameter values from the iteration of the terminated run.

Increasing the step-cut limit beyond 4 or 5 is generally not helpful but may be tried.

INV ERRØR DETERMINANT OF A=0 RANK OF A=K

The program attempted to invert a singular matrix. The two most frequent causes are a singular inertia matrix in subroutine STATE or a singular information matrix in subroutine SMAIN. Both are the result of errors in the program's input. The information matrix will be singular if, for the vector of measurement estimates $\hat{y}$ and for some one identified parameter θ_j, $\partial\hat{y}/\partial\theta_j \equiv 0$ for the entire time history. For example, if the user mistakenly tries to identify the bias in the roll attitude gyro, b_ϕ, but roll attitude is not one of the measurements, then $\partial\hat{y}/\partial b_\phi \equiv 0$.

(2) either the relative or absolute error bound specified in card type 2 (Table 4.1) is negative, or both are zero.

*** ERRØR IN SUBRØUTINE UPDATE. TØØ MANY INCREASES IN THE INTEGRATIØN ERRØR TØLERANCES.

If the program cannot integrate the differential equations within the error tolerances specified by the user, it increases them, prints a message and continues. If this occurs twice, the program terminates. Corrective action: examine input error tolerances for the possibility of increase or examine data for error.

*** ERRØR IN SUBRØUTINE UPDATE. MØRE THAN MAXNUM STEPS NEEDED BETWEEN T = x AND T = y.

The program's integration algorithm has determined that in order to meet the error bounds it must take more steps between times x and y than the maximum specified in input card type 3 (Table 4.1). Corrective action:

(1) verify that the input parameter values are reasonable, and if they are, then

(2) increase the number of steps permitted, or

(3) relax the error bounds.

*** EQUATIØNS APPEAR TØ BE STIFF

There are almost certainly errors in the input to the program. However, the differential equations could conceivably appear stiff because of exceptionally noisy control variable time histories.

*** MAXIMUM NUMBER ØF ITERATIØNS EXCEEDED.

The Levenberg-Marquardt search has not converged in the maximum number of iterations specified in card type 2 (Table 4.1). Corrective action: re-examine convergence criterion in input data and adjust as necessary.

7.3 ERRORS NOT CAUSING TERMINATION

Sometimes errors are detected which are not critical to the program's computation, and it prints a message and continues. The conditions resulting in these error messages are described below.

*** ERRØR IN INPUT CARD DECK WHILE ATTEMPTING TØ READ A LABEL. ERRØR ØCURRED ØN THIS CARD:

The character in the first column of an input card of type 7 (Table 4.1) was invalid. The program skips this card and continues to read the next card, which it also expects to be a card of type 7.

*** INTEGRATIØN ERRØR TØLERANCES ARE TØØ SMALL TØ CØNTINUE. THEY HAVE BEEN GIVEN NEW VALUES AT SAMPLE PØINT j. RELERR = x ABSERR = y.

The program has determined that it cannot meet the error bounds specified by the user and replaces them with the smallest bounds it believes it can meet.

*** ERRØR. A NEGATIVE EIGENVALUE, x, WAS CØMPUTED FØR THE INFØRMATIØN MATRIX. IT HAS BEEN SET = 1.E-27.

Because the information matrix is theoretically positive semidefinite, none of its eigenvalues should be negative. A slightly negative eigenvalue occasionally results from numerical error when it should actually be slightly positive. A large negative eigenvalue invalidates the run, and a serious error exists.

7.4 CONTROL CARDS

Figure 7.1 shows a typical set of control cards needed to run NLSCIDNT. The job and accounting cards are installation dependent and so are not shown in detail. The syntax of the other control cards may also be installation dependent; the cards shown are for the SCOPE 2.1 operating system on the CDC 7600 computer.

Tape 2 is the flight test or simulation data to be used as input. Tape 3 is the output generated by NLSCIDNT (see Section 5.2); if the output does not need to be saved, remove the REQUEST and CATALOG cards for tape 3.

```
JOB CARD
ACCOUNT CARD

    Other cards required by installation to
    initiate a job.

ATTACH,     TAPE 2 = RSRA3, ID = RSH
REQUEST,    TAPE 3, *PF
ATTACH,     NLSCIDNT, ID = BJB
NLSCIDNT,   PL = 7777
CATALOG,    TAPE 3 = RSRANL, ID = RSH.

    7/8/9      (End of Record)

    INPUT DATA CARDS (see Sections 4.1 & 4.2)

    6/7/8/9    (End of File)
```

Figure 7.1 Control Cards

REFERENCES

1. Hall, Jr., W.E., Gupta, N.K., and Smith, R.G., "Identification of Aircraft Stability and Control Derivatives for the High Angle-of-Attack Regime," Technical Report No. 2, prepared for the Office of Naval Research under Contract N00014-72-C-0328, March 1974.

2. Gupta, N.K. and Mehra, R.K., "Computational Aspects of Maximum Likelihood Estimation and Reduction in Sensitivity Function Calculation," IEEE Transactions on Automatic Control, Vol. AC-19, No. 6, December 1974.

3. Shampine, L.F. and Gordon, M.K., Computer Solutions of Ordinary Differential Equations, W.H. Freeman and Co., San Francisco, 1975.

4. Taylor, L., et al., "A Comparison of Newton-Raphson and Other Methods for Determining Stability Derivatives from Flight Data," Third Technical Workshop on Dynamic Stability Problems, Ames Research Center, 1968; also presented at AIAA Third Flight Test, Simulation and Support Conference, Houston, Texas, March 1969.

5. Bellman, R., et al., "Quasilinearization, System Identification and Prediction," RAND Corporation, RM-3812, August 1963.

6. Fisher, R.A., Statistical Methods and Scientific Inference, Oliver and Boyd, Edinburg, 1956.

7. Fisher, R.A., "Two New Properties of the Mathematical Likelihood," Proc. Roy. Soc.,London, 144, 285; 1934.

8. Barnard, G.A., "Statistical Inference," Journal Royal Stat. Society, B11, 116, 1949.

9. Kailath, T., "A General Likelihood-Ratio Formula for Random Signals in Gaussian Noise," IEEE Trans. Info. Theory, Vol. IT-15, May 1969.

10. Hall, W. E., Jr., Bohn, J. G., and Vincent, J. H., "Development of Advanced Techniques for Rotorcraft State Estimation and Parameter Identification", NASA CR-159297, August 1980.

APPENDIX A

MAXIMUM LIKELIHOOD IDENTIFICATION OF PARAMETERS OF A NONLINEAR SYSTEM*

A.1 INTRODUCTION

The maximum likelihood method is one of the most flexible techniques in statistics for identification of parameters from input-output data. Suppose it is possible to make a set of observations on a system, whose model has m unknown parameters θ. For any given set of values of the parameters θ from the feasible set Θ, we can assign a probability $p(Z|\theta)$ to each outcome Z. If the outcome of an actual experiment is z, it is of interest to know which sets of values of θ might have led to these observations. This concept is embedded in the <u>likelihood function</u> $\mathscr{L}(\theta|z)$. This function is of fundamental importance in estimation theory because of the likelihood principle of Fisher and others [6 - 8] which states that if the system model is correct, all information about unknown parameters is contained in the likelihood function. The maximum likelihood method finds a set of parameters $\hat{\theta}$ to maximize this likelihood function

$$\hat{\theta} = \max_{\theta \varepsilon \Theta} \mathscr{L}(\theta/z) \qquad \text{(A.1)}$$

In other words, the probability of the outcome z is higher with parameters $\hat{\theta}$ in the model than with any other values of parameters from the feasible set. Usually it is more convenient to work with the logarithm of the likelihood function (it is possible to do so because the logarithm is a strictly monotonic function).

---

* This appendix is a shortened version of Appendix A in Ref. 1. Only portions applicable to the NLSCIDNT program are included to provide a ready reference for the NLSCIDNT user.

The great asset of the maximum likelihood method is that it can be used with linear or nonlinear models in the presence of process and measurement noise. The maximum likelihood estimates are asymptotically unbiased, consistent and efficient. These terms are defined below.

(1) Bias: $\hat{\theta}$ is an unbiased estimator of θ if

$$E(\hat{\theta}) = \theta \tag{A.2}$$

where E stands for the "expected value." In physical terms, this implies that if the data were collected a large number of times, the mean of the different estimates $\hat{\theta}$ is θ.

(2) Efficient: $\hat{\theta}$ is an efficient estimator of θ if it is unbiased and if for any other unbiased estimator $\tilde{\theta}$ of θ

$$E(\hat{\theta} - \theta)^2 \leq E(\tilde{\theta} - \theta)^2 \tag{A.3}$$

(3) Consistent: $\hat{\theta}$ is a consistent estimator of θ if the accuracy of the estimate improves with increasing amount of data and approaches the true value as the amount of data tends to infinity. In other words, for any positive η and ε

$$P(|\hat{\theta}_n - \theta| < \varepsilon) > 1 - \eta \qquad n > N \tag{A.4}$$

This definition is analogous to the definition of convergence in real analysis.

A.2 GENERAL NONLINEAR SYSTEMS

Consider the general nonlinear aircraft equations of motion (without process noise)

$$\dot{x} = f(x,u,\theta,t) \qquad 0 \leq t \leq T \tag{A.5}$$

where

$x(t)$ is $n \times 1$ state vector

$u(t)$ is $\ell \times 1$ input vector

θ is $m \times 1$ vector of unknown parameters

Sets of p measurements $y(t_k)$ are taken at discrete times t_k

$$y(t_k) = h(x(t_k), u(t_k), \theta, t_k) + v(t_k)$$

$$k = 1,2,3,\ldots,N \tag{A.6}$$

$v(t_k)$ is Gaussian random noise with the following properties

$$E[v(t_k)] = 0$$

$$E[v(t_j)v^T(t_k)] = R(\theta, t_j)\delta_{jk} \tag{A.7}$$

The unknown parameters are supposed to occur in the functions f and h and in matrices R and x_o. In the analysis to follow, the model and the functional form of f and h is assumed known correctly.

The set of observations $y(t_1), y(t_2), \ldots, y(t_N)$ constitutes the outcome Z in this case. The likelihood function for θ, which has the same form as the probability of the outcome z for a certain value of parameters θ, is given by

$$\mathscr{L}(\theta|z) \tilde{=} p(z|\theta)$$

$$= p(y(t_1),\ y(t_2),\ \ldots,\ y(t_N)|\theta)$$

$$= p(Y_N|\theta) \tag{A.8}$$

where

$$Y_k = \{y(t_1),\ldots,y(t_k)\}, \quad k = 1,2,\ldots,N$$

$$p(Y_N|\theta) = p(y(t_N)|Y_{N-1},\theta)\cdot p(Y_{N-1}|\theta)$$

$$= p(y(t_N)|Y_{N-1},\theta)\ p(y(t_{N-1})|Y_N-2,\theta)\ p(Y_{N-2}|\theta)$$

$$= \prod_{i=1}^{N} p(y(t_i)|Y_{i-1},\theta) \tag{A.8}$$

The log-likelihood function is

$$\log\ [\mathscr{L}(\theta|z)] = \sum_{i=1}^{N} \log\ \{p(y(t_i)|Y_{i-1},\theta)\} + \text{constant} \tag{A.9}$$

To find the probability distribution of $y(t_i)$ given Y_{i-1} and θ, the mean value and covariance are determined first.

$$E(y(t_i)|Y_{i-1},\theta)\ \underline{\Delta}\ \hat{y}(i/i-1) \tag{A.10}$$

The expected value or the mean is the best possible estimate of measurements at a point given the measurements until the previous point.

$$\text{cov}(y(t_i)|Y_{i-1},\theta) = E\{[y(t_i)-\hat{y}(i/i-1)][y(t_i)-\hat{y}(i/i-1)]^T\}$$

$$\underline{\Delta}\ E\{\nu(i)\ \nu(i)^T\}$$

$$\underline{\Delta}\ B(i) \tag{A.11}$$

$\nu(i)$ are the innovations at point i and $B(i)$ is the innovations covariance. Since

$$y(t_i) - E(y(t_i)|Y_{i-1},\theta) = \nu(i) \tag{A.12}$$

it follows that $y(t_i)$ given Y_{i-1} and θ have the same distribution as $\nu(i)$. Kailath [9] has shown that as the sampling rate is increased, the innovations $\nu(i)$ tend towards having a Gaussian density. Assuming a sufficiently high sampling rate, the distribution of $\nu(i)$ and, therefore, $y(t_i)$ given Y_{i-1} and θ is Gaussian, i.e.,

$$p(y(t_i)|Y_{i-1},\theta) \cong \frac{\exp\{-\frac{1}{2}\nu(i)^T B^{-1}(i)\nu(i)\}}{(2\pi)^{m/2}|B(i)|^{1/2}} \tag{A.13}$$

$$\log p(y(t_i)|Y_{i-1},\theta) = -\frac{1}{2}\nu(i)^T B^{-1}(i)\nu(i) - \frac{1}{2}\log|B(i)| + \text{constant} \tag{A.14}$$

The log-likelihood function of Eq. (A.9) can be written as

$$\log[\mathscr{L}(\theta|z)] = -\frac{1}{2}\sum_{i=1}^{N}\{\nu^T(i)B^{-1}(i)\nu(i) + \log|B(i)|\} \tag{A.15}$$

An estimate of the unknown parameters is obtained by maximizing the likelihood function or the log-likelihood function from the feasible set of parameter values.

$$\hat{\theta} = \max_{\theta\varepsilon\Theta} \log[\mathscr{L}(\theta|z)] \tag{A.16}$$

$$= \max_{\theta\varepsilon\Theta} [-\frac{1}{2}\sum_{i=1}^{N}\{\nu^T(i)B^{-1}(i)\nu(i) + \log|B(i)|\}] \tag{A.17}$$

The log-likelihood function depends on the innovations and their covariance. To optimize the likelihood function, a way must be found for determining these quantities. In the absence of process noise, this is a simple task.

The state prediction is done using the equations of motion (A.5). The innovation is

$$\nu(i) = y(t_i) - h(\hat{x}(t_i), u(t_i), \theta, t_i) \tag{A.18}$$

From Eqs. (A.6) and (A.7) it can be seen that the covariance of the innovations is

$$B(i) = R(\theta, t_i) \tag{A.19}$$

A.3 OPTIMIZATION PROCEDURE

Many possible numerical procedures can be used for this optimization problem. Modified Newton-Raphson [1,2] or Quasilinearization [4] have been found by experience to give quicker convergence than most procedures like the conjugate gradient or the Davison method. The modified Newton-Raphson is a second order gradient procedure requiring computation of first and second order partials of the log-likelihood function, which, after substituting Eq. (A.19) into (A.15), is

$$\log\,[\mathscr{L}(\theta|z)] = -\frac{1}{2}\sum_{i=1}^{N} \nu^T(i)R^{-1}\nu(i) + \log|R| \tag{A.20}$$

$$\frac{\partial \log [\mathscr{L}(\theta|z)]}{\partial \theta_j} = -\sum_{i=1}^{N} \left\{ \nu^T(i)R^{-1}(i) \frac{\partial \nu(i)}{\partial \theta_j} - \frac{1}{2} \nu^T(i)R^{-1}(i) \cdot \frac{\partial R(i)}{\partial \theta_j} R^{-1}(i)\nu(i) + \frac{1}{2} \mathrm{Tr}\left(R^{-1}(i) \frac{\partial R(i)}{\partial \theta_j}\right) \right\} \tag{A.21}$$

$$\begin{aligned}
\frac{\partial^2 \log [\mathscr{L}(\theta|z)]}{\partial \theta_j \partial \theta_k} = -\sum_{i=1}^{N} \Bigg\{ & \frac{\partial \nu^T(i)}{\partial \theta_k} R^{-1}(i) \frac{\partial \nu(i)}{\partial \theta_j} \\
& - \frac{\partial \nu^T(i)}{\partial \theta_k} R^{-1}(i) \frac{\partial R(i)}{\partial \theta_j} R^{-1}(i)\nu(i) \\
& - \frac{\partial \nu(i)}{\partial \theta_j} R^{-1}(i) \frac{\partial R(i)}{\partial \theta_k} R^{-1}(i)\nu(i) \\
& + \nu^T(i)R^{-1}(i) \frac{\partial R(i)}{\partial \theta_j} R^{-1}(i) \frac{\partial R(i)}{\partial \theta_k} R^{-1}(i)\nu(i) \\
& - \frac{1}{2} \mathrm{Tr}\left[R^{-1}(i) \frac{\partial R(i)}{\partial \theta_j} R^{-1}(i) \frac{\partial R(i)}{\partial \theta_k}\right] \\
& + \frac{\partial^2 \nu(i)}{\partial \theta_j \partial \theta_k} R^{-1}(i)\nu(i) \\
& - \nu^T(i) R^{-1}(i) \frac{\partial^2 R(i)}{\partial \theta_j \partial \theta_k} R^{-1}(i)\nu(i) \\
& + \frac{1}{2} \mathrm{Tr}\left(R^{-1}(i) \frac{\partial^2 R(i)}{\partial \theta_j \partial \theta_k}\right) \Bigg\} \qquad j,k = 1,2,\ldots,p
\end{aligned} \tag{A.22}$$

The last three terms in the equation for second partial of the log-likelihood function involve second partials of innovation and its covariance. Those terms are dropped because they are much smaller than the other terms near a minimum and they are expensive to compute.

Solving Eq. (A.21) for unknown parameters in R, when R is not a function of time, gives

$$\hat{R} = \frac{1}{N} \sum_{i=1}^{N} \nu(i)\, \nu^T(i) \tag{A.23}$$

The equality in (A.23) holds only for those elements of R which are not known a priori. For instance, even if R is known to be diagonal, the right hand side matrix will not be diagonal in general, but the off-diagonal terms should be ignored before they are equated to R. Using (A.23) in (A.20)

$$\log[\mathscr{L}(\theta|z)] = -\frac{1}{2} \sum_{i=1}^{N} \nu^T(i)\, \hat{R}^{-1} \nu(i) + \text{constant} \tag{A.24}$$

The first and second derivatives of the log-likelihood function with respect to unknown parameters (which do not appear in R) are

$$\frac{\partial}{\partial\theta_j} \log[\mathscr{L}(\theta|z)] = -\sum_{i=1}^{N} \nu^T(i)\, \hat{R}^{-1} \frac{\partial\nu(i)}{\partial\theta_j} \tag{A.25}$$

$$\frac{\partial^2 \log[\mathscr{L}(\theta|z)]}{\partial\theta_j \partial\theta_k} = -\sum_{i=1}^{N} \left\{ \frac{\partial\nu^T(i)}{\partial\theta_k} \hat{R}^{-1} \frac{\partial\nu(i)}{\partial\theta_j} + \nu^T(i)\, \hat{R}^{-1} \frac{\partial^2\nu(i)}{\partial\theta_j \partial\theta_k} \right\} \tag{A.26}$$

The terms in the second derivative are approximated as

$$\frac{\partial^2 \log[\mathscr{L}(\theta|z)]}{\partial\theta_j \partial\theta_k} = -\sum_{i=1}^{N} \left\{ \frac{\partial\nu^T(i)}{\partial\theta_k} \hat{R}^{-1} \frac{\partial\nu(i)}{\partial\theta_j} \right\} \tag{A.27}$$

One Newton-Raphson iteration evaluates the first and second gradients for the parameter values given it and computes new parameter values as follows

$$\theta_{new} = \theta_{old} + \Delta\theta \tag{A.28}$$

$$\Delta\theta = -\left(\frac{\partial^2 J}{\partial\theta^2}\right)^{-1} \left(\frac{\partial J}{\partial\theta}\right)^T \tag{A.29}$$

where

$\Delta\theta$ = the step vector,

$J = -\log[\mathscr{L}(\theta|z)]$,

$\frac{\partial J}{\partial\theta}$ = $1 \times m$ row vector whose j-th element is $\partial J/\partial\theta_j$, and

$\frac{\partial^2 J}{\partial\theta^2}$ = $m \times m$ symmetric matrix whose (j,k)th element is $\partial^2 J/(\partial\theta_j \partial\theta_k)$.

Because $\partial^2 J/\partial\theta^2$ is real and symmetric,

$$\left(\frac{\partial^2 J}{\partial\theta^2}\right)^{-1} = \sum_{i=1}^{m} \frac{1}{\lambda_i} \xi_i \xi_i^T \tag{A.30}$$

where

λ_i is the i-th eigenvalue of $\partial^2 J/\partial\theta^2$, and

ξ_i is the i-th eigenvector.

Let $g \triangleq \left(\frac{\partial J}{\partial \theta}\right)^T$ for simplicity of notation; then

$$\Delta\theta = -\sum_{i=1}^{m} \frac{1}{\lambda_i} \xi_i \xi_i^T g = -\sum_{i=1}^{m} \left(\frac{\xi_i^T g}{\lambda_i}\right) \xi_i \qquad \text{(A.31)}$$